
U N I V E R S I T Y  O F  L I E G E

*Aerospace and Mechanical Engineering Department*

# Generalized Shape Optimization using XFEM and Level Set Description

PhD Thesis Dissertation
submitted in fulfillment of the requirements for the degree of
'Docteur en Sciences de l'Ingénieur'

by

**Laurent Van Miegroet**
Ingénieur Civil Electro-Mécanicien (Aérospatiale)

March 2012

**Members of the Examination Committee**

Jean-Philippe Ponthot (President of the committee)
Professor - Université de Liège

Pierre Duysinx (Supervisor)
Professor - Université de Liège

Eric Béchet
Professor - Université de Liège

Christophe Geuzaine
Professor - Université de Liège

Jean-François Remacle
Professor - Université catholique de Louvain

Franz Josef Barthold
Professor - Technische Universität Dortmund, Germany

Michael Yu Wang
Professor - The Chinese University of Hong Kong, P.R. China

Michael Bruyneel
PhD, Head of Analysis Group - LMS-Samtech

# Abstract

CAD based shape optimization aims at finding the shapes of internal and external boundaries of a structural components. The method is able to improve the design of structures against various criteria such as restricted displacements, stress criteria, eigenfrequencies, etc. However, this technique has been quite unsuccessful in industrial applications because of the mesh management problems coming from the large shape modifications. The main technical problems stems from the sensitivity analysis requiring the calculation of the so-called velocity field related to mesh modifications. If 2D problems are quite well mastered, 3D solid and shell problems are difficult to handle in the most general way. It turns out that shape optimization remains generally quite fragile and delicate to use in industrial context. To circumvent the technical difficulties of the moving mesh problems, a couple of methods have been proposed such as the fictitious domain approach, the fixed grid finite elements and the projection methods.

The present work relies on the application of the extended finite element method (X-FEM) to handle parametric shape optimization. The X-FEM method is naturally associated with the Level Set description of the geometry to provide an efficient and flexible treatment of problems involving moving boundaries or discontinuities. On the one hand, the method proposed benefits from the fixed mesh approach using X-FEM to prevent from mesh management difficulties. On the other hand, the Level Set description provides a smooth curves representation while being able to treat topology modifications naturally.

In this thesis, we focus on the material-void and bi-material X-FEM elements for mechanical structures. The representation of the geometry is realized with a Level Set description. Basic shapes can be modeled from simple Level Set such as plane, circle, ... NURBS curves and surfaces that can be combined together using a Constructive Solid Geometry approach to represent complex geometries. The design variables of the optimization problem are the parameters of basic Level Set features or the NURBS control points. Classical global (compliance, eigenfrequencies, volume) and local responses (such as stress constraint) can be considered in the optimization problem that is solved using a mathematical programming approach with the CONLIN optimizer. The problem of the computation of the shape sensitivity analysis with X-FEM is carefully addressed and investigated using several original methods based on semi-analytical and analytical approaches that are developed. Academic examples are first considered to illustrate that the proposed method is able to tackle accurately shape optimization problems. Then, real life structures including 2D and 3D complex geometries illustrate the advantages and the drawbacks of using X-FEM and Level Set description for generalized shape optimization.

# Acknowledgements

*I would like to express my deepest gratitude for all the support, advice, instruction, guidance and patience of my advisor Professor Pierre Duysinx who gave me the opportunity to realize this research. It has been a profound pleasure to work and travel with him during these past years. Thank you also for the meticulous reading of the thesis draft. I would like to express my appreciation to Michael Bruyneel for his precious help.*

*Equally, I would like to thank Thibault Jacobs for introducing me to the OOFELIE software and for kindly sharing his C++ expertise. Furthermore, I should mention that some parts of this work has been realized with the helpful collaboration of Thibault Jacobs, Véronique Rochus, Séverine Lepage and Thibaut Schoonjans.*

*I acknowledge all the staff of the Aerospace and Mechanics department for their valuable camaraderie. I wish them great success in their research and family life. Especially, i would like to thank Etienne Lemaire, my office mate for so many years, for all the enjoyable days we had together.*

*I thank my family and especially Caroline for all her emotional support, patience and encouragement during the redaction of this thesis.*

*This research was developed along two funded research projects and i gratefully acknowledge the support of the project ARC MEMS, Action de Recherche Concertée 03/08-298 funded by the Communauté Française de Belgique and the Convention n° 616420 EFCONIVO of the Programme WIST 2 from the Région Wallonne.*

# Contents

Chapter 1

# Introduction

Nowadays the domain of structural design remains a complex and arduous task for engineers as they have to face every day new challenging problems. Hence, whatever the field of his activity, the designer must continuously create new innovative solutions to achieve better performance while satisfying the demands from the industry and markets.

In high technology industries such as aerospace and automotive, the engineer has to present high quality solutions. For instance, the design of a structure must satisfy numerous requirements considering the mass of the structure, its global resistance, its cost of development and also manufacturing and practical aspects.

Generally the best solution results from a trade off between the various design criteria imposed by the manufacturing specifications which are very often conflicting. Such antagonistic criteria arise in the typical example of the design of a structure which should have the smallest mass while retaining the strongest mechanical resistance.

The traditional design strategy considers a previous existing design as a starting point to create a new structure. The intuition of the engineer lead to step by step modifications from an initial design towards a solution which should exhibits a better performance. However, this trials-and-errors procedure, which implies, in case of unsatisfactory solution, to investigate other concepts, is both time consuming and expensive. Furthermore, the solution obtained is generally very similar to the initial design and there is no guarantee that this is an optimal solution. While this approach has been applied for several decades, the increasingly more complex problems that engineers have to solve now make the design process so difficult that new approaches are desirable. For example, the drastic rise of fuel price during the last 10 years have forced the aerospace and automotive industries to resort to modern optimization techniques to achieve significant weight reductions.

Beside the development of numerical methods in engineering to extend analysis capabilities, the research has also focused on the development of automatic design methods. The automation of the design procedure leads naturally to develop adapted procedures in which computers could greatly help the engineer in the search for the optimal design. Depending on the nature of the design variables, different classes of structural optimization problems have been addressed: the sizing, the shape and the topology optimization.

## 1.1 The hierarchy of structural optimization problems

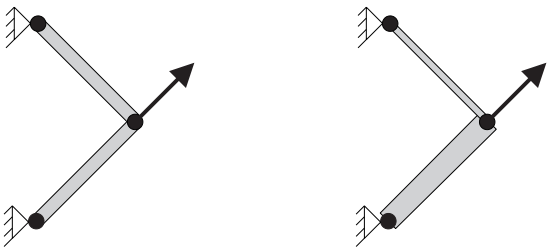


Figure 1.1: Illustration of a sizing optimization problem

The first kind of optimization problem which has been investigated is the *optimal sizing* of structures. This is the simplest strategy in structural optimization. In this approach, only the transverse dimensions of structures, as illustrated in Figure 1.1, such as cross section of bars and beams or the thickness of plates and shells are modified. Therefore, no modifications of the shape or the topology is involved and the finite element mesh does not change during the redesign process. This method is now well established and is able to treat industrial applications with a gain of 5 % to 10 % in performance. The class of problem studied with this method are truss-like or frames structures.

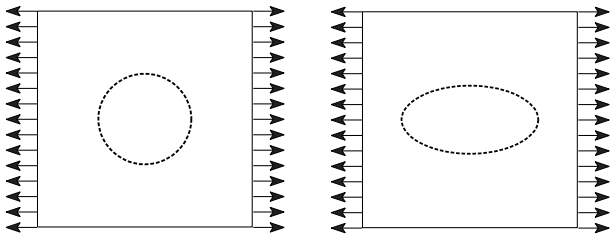


Figure 1.2: Illustration of a shape optimization problem

*Shape optimization* has now been investigated for more than 30 years. It aims at finding the shapes of internal and external boundaries without modifying the structural topology as in Fig. 1.2. The method has a great interest in the improvement of the detailed design of structures against various criteria such as restricted displacements, stress criteria, buckling, etc and it is also able to treat multidisciplinary or multiphysics problems. Two different approaches can be considered to modify the geometry. As suggested by Braibant and Fleury [38], a regular formulation of the shape optimization problem can be defined at the level of the CAD model independently of the finite element model. The design variables are then selected among the control points or the parameters describing the geometrical entities. Or, as proposed by Bletzinger *et al.* [33], one can also work directly on FE-based data by considering the nodal coordinates and nodal thickness as design variables. The major difficulty of the shape optimization formulation is related to the mesh management problems coming from the large shape modifications. If 2D problems are quite well mastered, 3D problems are still difficult to handle in the most general case due to problems related to mesh distortions or automatic and reliable mesh (re)generation difficulties. Moreover, large modifications of the shape parameters usually result to a situation in which it is impossible to reconstruct the geometric model without human intervention.

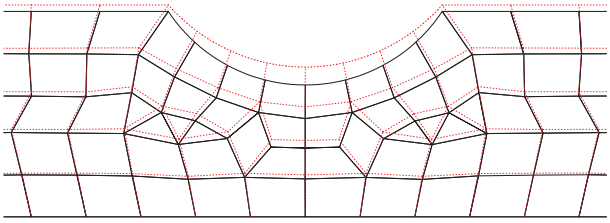


Figure 1.3: Illustration of the Velocity Field and mesh perturbation

Another main technical problem stems from the sensitivity analysis. Shape derivatives have to be regarded as material derivatives and the sensitivity analysis requires the calculation of the so-called Velocity Field [17, 18] or practically of a perturbed mesh as illustrated in Figure 1.3. Currently, it turns out that shape optimization remains generally quite fragile and delicate to use in industrial context with little application in practice nowadays.

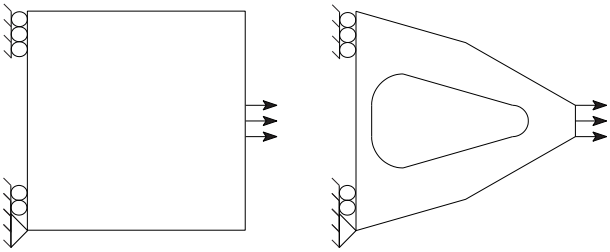


Figure 1.4: Illustration of a topology optimization problem

*Topology optimization* has been first handled efficiently with the homogenization method by Bendsøe and Kikuchi [28] and is now available within several commercial finite element codes [11, 146]. The optimization method is formulated as an optimal material distribution problem [29]. A given amount of material is distributed in a design domain in order to minimize some performance criteria, generally a global criterion such as the compliance or the fundamental eigenfrequency. As illustrated in Figure 1.5, by adding or removing material, one is able to determine the optimal structure inside a prescribed design domain without any a priori knowledge of the solution.

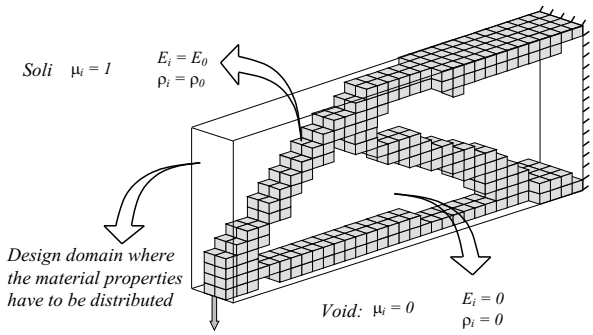



Figure 1.5: Parametrization of a topology optimization problem

Practically, one major advantage of the formulation is that it works on a fixed mesh. The design variables are the element or nodal densities. This parameterization allows topological modifications, deep geometrical remodeling providing large gains of performance. The drawback of this formulation comes from the very large scale of the optimization problem with thousands of design variables. Therefore one generally considers simple design problems as the minimum compliance problem with a single volume constraint. Introducing local stress constraints as in Duysinx and Bendsøe [58] leads to huge problems, which may become difficult to handle, whereas the physical meaning of the local stress constraints can become questionable because of the jumps of density between neighboring elements and the absence of geometric description. In addition density formulation makes quite sophisticated any control of the geometry (minimum size, perimeter, etc.), which is mainly related to manufacturing considerations. Finally the definition of the optimal structure as a picture with gray material and chattered boundaries fosters engineers to interpret the optimal design to later reconstruct a parametric CAD model. This interpretation step sometimes degrades strongly the performance of the structure [131].

## 1.2 Context and objectives of this thesis

For the last 20 years shape optimization has been stocking to penetrate industrial and real-life applications, whereas topology optimization has experienced a fast growing soar and was extended to various applications. By the way there is still a great interest in shape optimization because of the intrinsic capability of shape description to consider easily complex problems involving local stress and manufacturing constraints for instance.

In order to circumvent the technical difficulties of the moving mesh problems, a couple of researches have tried to formulate shape optimization using different numerical methods and parameterizations. Among others, we can mention fixed mesh approaches such as the fictitious domains method of Dankova and Haslinger [50] or the projection methods by Norato *et al.* [127] and more recently the Eulerian shape optimization of Kim and Chang [93]. Non conventional methods like the Meshfree have also been applied to shape optimization in 2D and 3D [94] and a T-Spline finite element method [77] based on the isogeometric analysis by Hughes *et al.* [83].

The Level Set method proposed by Osher and Sethian [133] has opened new perspectives to handle variable and moving boundaries. The implicit description of the geometrical entities enables for a more friendly and flexible manipulation of the geometry and overcomes some restrictions related to the explicit approach of the geometry used in CAD systems. For example it is possible to reduce the topological complexity by removing or merging geometric entities (i.e. holes can merge or disappear) without degenerating the model, which is a major restriction of shape optimization using CAD systems. These advantages have been early identified and applied to topology optimization in the seminal works of Allaire *et al.* [9] and Wang *et al.* [177].

In the late 2000's a new fixed mesh method called *eXtended Finite Element Method* (X-FEM) has been proposed as an alternative to remeshing methods [114]. This approach greatly reduces the difficulty of considering time-variable boundaries and complex geometries by using non-conforming meshes. The X-FEM method is naturally associated with the *Level Set* description of the geometry to provide a very efficient treatment of difficult problems involving discontinuities and propagations. Up to now X-FEM has been mostly developed for crack propagation problems [114], but the potential interest of the X-FEM and the Level Set description for other problems like topology optimization was identified very early in Belytschko *et al.* [25].

In this thesis, we propose to use the X-FEM and the Level Set description as an elegant way to fill the gap between topology and shape optimization. The method can be qualified as *generalized shape optimization* as it presents smooth boundary descriptions while allowing topology modifications as holes can merge and disappear. X-FEM enables working on a fixed mesh, as in topology optimization, circumventing the technical difficulties of shape optimization. The structural shape description uses basic Level Set features (circles, rectangles, etc.) or NURBS curves transformed into a Level Set that can be freely combined to generate any shapes. The design variables are the parameters of the Level Set features or the NURBS control points.

This work extends and summarizes the investigations realized in preliminary studies on the application of X-FEM and Level Sets to shape optimization [173, 174]. In these works, we obtained promising results with a rather simple approach based on finite differences and semi-analytic approach in 2D. However, these works pointed out some specific aspects and difficulties that needed to be further investigated.

The first objective of the present Ph.D is to develop X-FEM elements for material-void interfaces in 2D and 3D along with a complete Level Set description of the geometry. The second objective lies in the development of a shape optimization framework adapted to the X-FEM and its shape sensitivity analysis. The finite difference approach used in the previous studies was to be replaced by an adapted and accurate semi-analytical approach. However, this method may be not adapted in all cases and flaws have been identified so several alternative strategies to circumvent these problems are proposed and validated.

All the developments of this thesis have been implemented in the C++ software OOFELIE [132].

## 1.3 Outline

The thesis outline is the following. The Level Set method is introduced in chapter 2. It mainly emphasizes on the geometrical representation aspects, referred here as Level Set description, rather than on the mathematical foundations of the Level Set method that are not used in this work.

Chapter 3 presents the basis of the extended finite element method for elastostatic analysis applied to three kinds of discontinuities whereas chapter 4 focuses on the details of efficient numerical implementation of the extended finite element method for the material-void interfaces.

The chapter 5 briefly describes the different approaches available to compute the sensitivity analysis in computational mechanic and particularly in the context of FEM based shape sensitivity analysis. The strategy is then extended to the X-FEM based shape optimization. Particular problem of shape optimization with extended elements are discussed in details and several approaches are proposed.

In chapter 6, academic and classical shape and topology optimization benchmarks are reinvestigated to illustrate the capabilities of the shape optimization method developed.

Finally, chapter 7 presents some possible future researches and extensions of the work. The present work is summarized in the conclusions in chapter 8.

Chapter 2

# Introduction to the Level Set Description of the geometry

The explicit representation of a structural shape using parametric CAD geometry makes difficult deep boundary or topological changes such as the creation of new holes, merging or separation of holes. This limitation is one of the main reasons of the low performances generally associated to the classical shape optimization approach [79]. However, the representation of the geometry using CAD curves has the advantage of keeping smooth boundaries clearly defined, which does not call for any interpretation as it is the case in topological optimization [131]. Therefore, a method that could present the advantage of the CAD representation without its drawbacks is of a great interest in shape optimization as well as in all kinds of problems in which topological changes or moving boundaries are involved.

The Level Set method, which has been initially developed by Sethian and Osher in 1987 [133], exhibits these two properties. The idea of this method is very simple and consists in representing the boundaries of the structure with an implicit function of a higher dimension. The main advantages of this representation is its ability to deal with topological changes while keeping smooth boundaries. Its implementation can be extended to any number of spatial dimensions without any difficulties and the numerical methods to solve the underlying equations are well established.

This method has been now successfully applied to a huge variety of problems ranging from multiphase fluid flows[167], fire propagation simulation [107], computer vision and shape recognition [91, 105], image processing [106] and even in movie special effect [59]. Moreover, as we will see later, the pairing of the Level Set Description/Method and the extended finite element method (X-FEM) is also particularly advantageous for tracking moving and evolving interfaces.

In this chapter, the general principles of the Level Set description and evolution method are introduced, as well as its advantages, its inherent drawbacks and the interesting complementarity with the X-FEM method. In this manuscript, a distinction is made between the Level Set Description which refers to the implicit geometrical representation and the Level Set Method which refers to the mathematical method for moving interfaces.

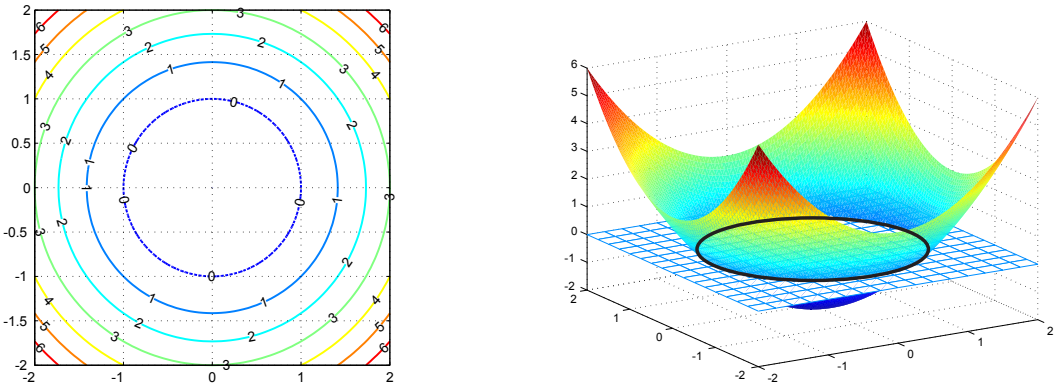
## 2.1 Description of the Level Set (Implicit representation)

The Level Set method is a numerical technique originally developed to analyze and follow the motions and deformations of an interface under an arbitrary velocity field. This velocity can depend: on time, on the position of the interface, on a related underlying physic problem, on a geometrical property of the interface or on any other parameters. While a usual geometric representation of the tracked interface generally rely on an explicit description, the key idea of Sethian and Osher [133] was to suggest the use of an implicit representation. They proposed to introduce a smooth scalar function $\phi(\mathbf{x})$ defined on all $\mathbf{x} \in R^n$ which, at all time, should represent an interface $\Gamma$ of dimension $n-1$ as the set where $\phi(\mathbf{x})$=0.

Hence, the interface is implicitly defined as the zero-level of a higher dimension scalar function $\phi(\mathbf{x})$. For example, if the interface $\Gamma$ corresponds to a circle of radius $r$, one could define the following function $\phi(\mathbf{x})$:

$$\begin{aligned} \phi(\mathbf{x}) &= x^2 + y^2 - r \\ \Gamma &= \left\{\mathbf{x} \in R^2 | \phi(\mathbf{x}) = 0\right\} \end{aligned}$$

rather than using the classical parametric definition in polar coordinates: $x$=$r\cos\theta$, $y$=$r\sin\theta$.



(a) $\phi(\mathbf{x})$=c contour representation

(b) $\phi(\mathbf{x})$ function

Figure 2.1: Implicit representation of a circular interface

The representation of $\Gamma$ and $\phi(\mathbf{x})$, with a radius $r = 1$ is depicted in Figure 2.1. The Figure 2.1 (a) represents different isocontours of the function $\phi(\mathbf{x})$. As expected, the shape of the isocontours are circular as the $\phi(\mathbf{x})$ function is in fact the equation of a circle in Cartesian coordinates and the iso-zero contour, which represents the interface (in dashed line), corresponds to a circle of radius $r = 1$. In Figure 2.1 (b), one can see the representation of the function $\phi(\mathbf{x})$ and its zero-level contour.

In practice, as an interface represented by an analytical expression is a quite restrictive case, the Level Set $\phi(\mathbf{x})$ can be constructed conveniently as a signed distance function. To define a signed distance function, we first introduce the distance function $d(\mathbf{x})$, which, on a domain $\Omega \in R^n$, is given as:

$$d(\mathbf{x}) = \min_{\mathbf{x_\Gamma} \in \Gamma} \|\mathbf{x} - \mathbf{x_\Gamma}\| \; \forall \mathbf{x} \in \Omega$$

where $\|.\|$ denotes the Euclidean norm and 'min' is the minimum operator. By its very definition, the value of the function $d$ is equal to zero if the point $\mathbf{x} \in \Gamma$ and $\|\mathbf{x} - \mathbf{x_\Gamma}\|$ otherwise, $\mathbf{x_\Gamma}$ being the closest point on the interface $\Gamma$ ($\mathbf{x_\Gamma} \in \Gamma$) to the point $\mathbf{x}$. For a circular interface of radius $r$, the corresponding distance function $d(\mathbf{x}) = \left|\sqrt{x^2 + y^2} - r\right|$, where $|\cdot|$ is the absolute value operator, is represented in Fig. 2.2.

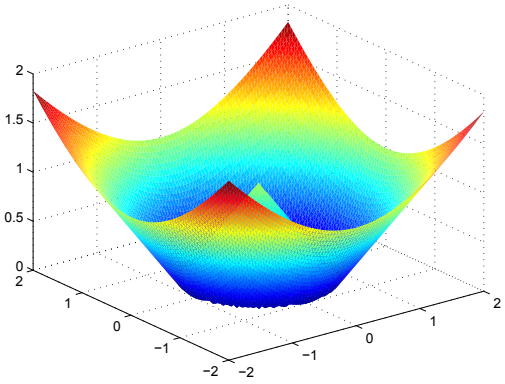
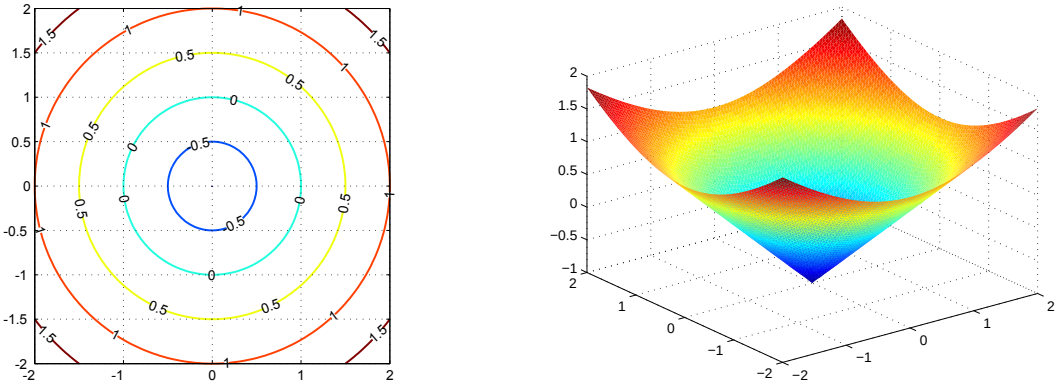


Figure 2.2: Representation of a circular interface with a distance function

Then, if we partition the domain $\Omega$ into $\Omega^+$, $\Omega^-$ and $\partial\Omega$ corresponding to the region outside the interface, inside the interface and on the interface $\Gamma$ ($\Gamma$=$\partial\Omega$) respectively, a signed distance function $\phi(\mathbf{x})$ to the interface is defined by $\phi(\mathbf{x}) = \pm \min_{\mathbf{x_\Gamma} \in \Gamma} \|\mathbf{x} - \mathbf{x_\Gamma}\|$ or:

$$\phi(\mathbf{x}) = \begin{cases} d(\mathbf{x}) & \text{if} \quad \mathbf{x} \in \Omega^+ \\ -\ d(\mathbf{x}) & \text{if} \quad \mathbf{x} \in \Omega^- \\ 0 & \text{if} \quad \mathbf{x} \in \partial\Omega \end{cases}$$



(a) $\phi(\mathbf{x})$=c contour representation

(b) $\phi(\mathbf{x})$ function

Figure 2.3: Level Set signed distance function representation of a circle

Hence, the implicit function $\phi(\mathbf{x})$ representing the circle of Fig. 2.1 can be equivalently replaced with the signed distance function $\phi = \sqrt{x^2 + y^2} - r$ representing the same unit radius ($r$=1)

circle interface $\Gamma$.

Given the fact that the distance function is always positive, the signed distance function has the following property:

$$\begin{aligned} \phi(\mathbf{x}) > 0 \quad &\text{if} \quad \mathbf{x} \in \Omega^+ \\ \phi(\mathbf{x}) = 0 \quad &\text{if} \quad \mathbf{x} \in \partial\Omega \\ \phi(\mathbf{x}) < 0 \quad &\text{if} \quad \mathbf{x} \in \Omega^- \end{aligned}$$

We can remark in Figure 2.3 that compared to the distance function, the signed distance function is smooth across the interface and can be differentiated without any problems. It also verifies the property $\|\nabla\phi(\mathbf{x})\| = 1$ which guarantees that it does not present too steep or too flat regions. When it is not the case, numerical inaccuracies can be encountered when localizing the iso-zero contour for instance.

## 2.2 Level Set method

Coming back to the original idea of the Level Set i.e. tracking the motion and the deformation of an interface, we now have to consider that both the interface $\Gamma$ and the scalar function $\phi(\mathbf{x})$ are dependent on time (or a pseudo time parameter): $\Gamma = \Gamma(t)$ and $\phi = \phi(\mathbf{x}, t)$. Moreover, for all time $t \geq 0$, the interface should always correspond to the set where the function $\phi(\mathbf{x}, t)$ vanishes: $\Gamma(t) = (\mathbf{x} | \phi(\mathbf{x}, t) = 0)$. Hence, as time increases and the interface $\Gamma$ propagates, $\phi(\mathbf{x}, t)$ should then evolve in such a way that the set of points $\phi(\mathbf{x}(\mathbf{t}), t) = 0$ represent the evolution of the interface $\Gamma(t)$.

Noting $\mathbf{x}(t)$ the successive positions of a point on the interface $\Gamma(t)$, $\mathbf{x}(t = 0)$ is a point on the initial interface and $\mathbf{x}(t = \tau)$ the position of the same material point at time $\tau$. Since the function $\phi(\mathbf{x}(t))$ should always be equal to zero on the interface, we have:

$$\phi(\mathbf{x}(t), t) = 0 \tag{2.1}$$

for all time $t$. Differentiating (2.1) using chain rule with respect to time gives:

$$\frac{\partial\phi(\mathbf{x}, t)}{\partial t} + \nabla\phi(\mathbf{x}(t), t) \cdot \frac{\partial \mathbf{x}(t)}{\partial t} = 0 \tag{2.2}$$

Now, if we note $\boldsymbol{F}$ the speed of the boundary in the outward normal direction, $\boldsymbol{N} = \frac{\nabla\phi}{\|\nabla\phi\|}$ the unit normal of $\phi$, and by $u$ the magnitude of the speed of the point $\mathbf{x}(t)$, $\frac{\partial \mathbf{x}(t)}{\partial t} = u$, we have: $\boldsymbol{F} = u \cdot \boldsymbol{N}$, and equation (2.2) can be rewritten as:

$$\begin{aligned} \frac{\partial\phi(\mathbf{x}, t)}{\partial t} + \boldsymbol{F} \cdot \|\nabla\phi\| &= 0 \\ \phi(\mathbf{x}, t) &= 0 \ \textit{given} \end{aligned}$$

Thus, the evolution of the boundary $\Gamma(t)$ is ruled by a first order hyperbolic PDE (Partial Differential Equation) for $\phi$, where $\phi(\mathbf{x}, t = 0)$ is the initial value of the problem. The evolution equation describing the propagation of the interface under a speed $u$ is turned into an equation involving the function $\phi(\mathbf{x}, t)$ and its normal vector field $\boldsymbol{F}$. From (2.3) one can remark that any tangential component of speed $u$ will have no effect on the position of the interface. This equation, due to the work of Osher and Sethian [133], has the form of a Hamilton-Jacobi equation

and is often referred as the *Level Set equation*.

Several techniques to solve the general equation of movement (2.3) of the Level Set have been developed. One of the most popular method is the high order Hamilton-Jacobi Essentially Non Oscillatory (ENO) method which is based on non oscillatory polynomial interpolation of data. However, the weighted version of the ENO scheme (WENO), proposed by Liu *et al.* [104] is considered as the state-of-the-art method to solve the level set equation. Interested readers can refer to [89, 134, 135] for a more exhaustive discussion about these methods.

In the seminal research on the Level Set Method, the level set function was initialized as an Heaviside distance function $\phi(\mathbf{x}) = 1 \pm d(\mathbf{x})$, where $d(\mathbf{x})$ is the distance to the interface $\Gamma$. However, numerical experiments have shown that it is obviously more appropriate to define $\phi$ as a smooth function. The most general implementation follows the proposition of Mulder, Osher and Sethian [124] to initialize the Level Set function as a signed distance function. Unfortunately, in practice, as the interface evolves, the $\phi$ function drifts away from a signed distance function. In this case, the $\phi$ function may become very steep or flat particularly near the interface which can cause numerical errors when computing derivatives or localizing the interface. In order to reduce these numerical errors, Chopp [48] proposed to periodically reinitialize the level set as a signed distance function. The most straightforward approach is to stop the iteration at some time $t$, locate the interface and rebuild a new Level Set function from this interface. However, this method presents the drawback of increasing the computational time.

Noting that only the isocontour of the Level Set is important, Chopp [48] proposed to efficiently reduce the cost of resolution by restricting the definition of the level set function to a neighborhood of the interface rather than defining it on the whole computational domain. Thus, reasonable computational times are obtained for the construction of a level set function in a band around the interface, the reinitialization procedure and the resolution of the level set equation. This technique, which presents a much lower operation cost than on the full domain $\Omega$, has been introduced as the *Narrow Band Level Set method* by Adalsteinsson and Sethian in [2]. Later, Sussman *et al.* [167] proposed to realize the reinitialization by solving a partial differential equation before updating the level set with equation (2.3):

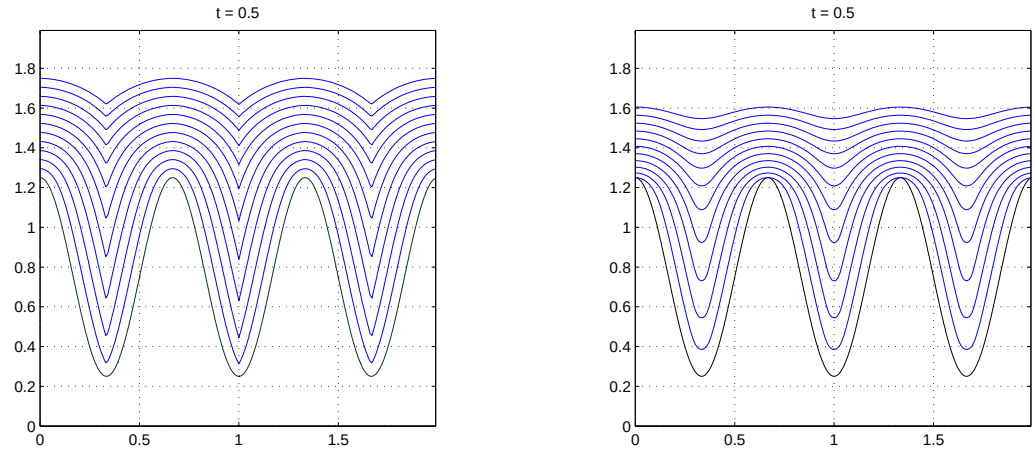
$$\begin{aligned} \frac{\partial \phi}{\partial t} + sign(\phi)(\|\nabla \phi\| - 1) &= 0 \\ \phi(\mathbf{x}, 0) &= \phi^{(t)} \end{aligned}$$

where $\phi^{(t)}$ is the level set function at time step $t$ of the level set equation. This equation is solved until a steady state is reached for which $\frac{\partial \phi}{\partial t} = 0$ and the signed distance property $\|\nabla \phi\| = 1$ holds. Although this method has the advantage of removing the need for locating the interface position, it also exhibits some drawbacks. The sign function has to be regularized which introduces some diffusion that results in inaccuracies when locating the interface.

The introduction of the $\phi(\mathbf{x})$ function of a higher dimension in $\Omega$ assumes that the velocity field $\boldsymbol{F}$ is also defined on the domain $\Omega$ and not only on the interface $\Gamma$. While it can be the case or at least in a neighborhood of the interface, $\boldsymbol{F}$ generally comes from the underlying physics of the problem and is only known along the interface $\Gamma$. Thus, $\boldsymbol{F}$ needs to be appropriately extended to the entire domain $\Omega$ from a prescribed speed $\boldsymbol{F}_\Gamma$ on the interface. Generally, the velocity is extended in such a way that the signed distance function property of $\phi(\mathbf{x})$ is preserved during

iteration i.e. with a speed $\boldsymbol{F}$ that verifies $\nabla \boldsymbol{F} \cdot \nabla \phi = 0$ (see [3] for more details).

To illustrate the type of solution that can be obtained with the Level Set method, we present in Figure 2.4 the evolution of 2 curves.




(a) Evolution with a speed function $\boldsymbol{F}$=1

(b) Evolution with a speed function $\boldsymbol{F}$=1-$\epsilon\kappa$

Figure 2.4: Evolution of 2D interfaces

In Figure 2.4 (a), a normal speed $\boldsymbol{F} = 1$ is applied to the initial curve in black, while the center one 2.4 (b) presents the evolution with a speed function $\boldsymbol{F}$ = 1-$\epsilon\kappa$ where $\kappa$ is the curvature of the curve and with $\epsilon$=0.05. In Fig. 2.4, we note that the curvature has a smoothing effect on the evolution of interface as the successive interfaces remain smooth while the left Figure develops sharp corners. The Figure 2.5 represents the evolution of a star-shaped interface under a constant normal motion. In these Figures, and specifically in the Figure 2.4 (a) and 2.5, there is no problem when the interface tends to get closer to itself.

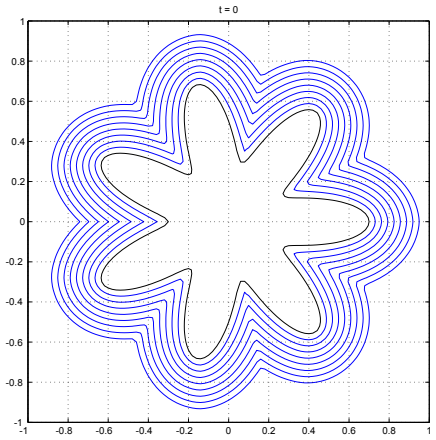



Figure 2.5: Evolution of 2D star-shaped interfaces

The Level Set Equation describes the evolution of an interface $\Gamma$ by monitoring the evolution of the implicit function $\phi$. Thus, the formalism is Eulerian and the interface is recovered by seeking

the iso-zero curve of $\phi$ at any time $t$. Hence, there is no difficulty to treat a modification of the topology like a separation as it can be easily noted in the Fig. 2.6 that represents the evolution of a dumbbell under a velocity field proportional to the mean curvature of the interface.

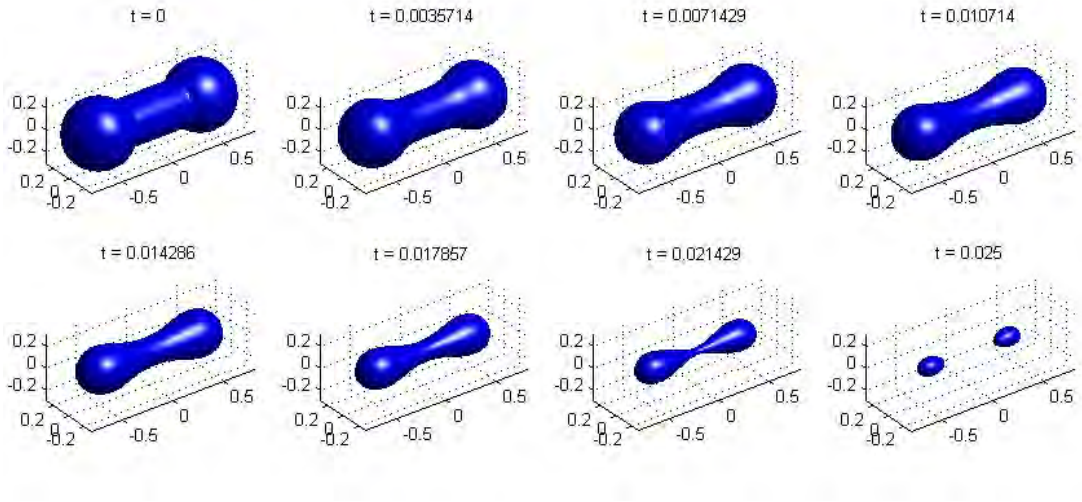



Figure 2.6: Level Set evolution of a dumbbell under mean curvature velocity field. Realized with the Ian Mitchell Level Set Matlab Toolbox [111]

In a Lagrangian approach, we should recover the positions of the interface points $\mathbf{x}(t)$ during the evolution of the interface by solving the following ordinary differential equation:

$$\frac{d\mathbf{x}}{dt} = \boldsymbol{F}(\mathbf{x}) \tag{2.3}$$

where $\boldsymbol{F}(\mathbf{x})$ is the velocity of a point $\mathbf{x}$ on the interface. As the interface is generally composed of an infinite set of points $\mathbf{x}$, in a discrete 2D setting, $\Gamma$ would be discretized into a finite number of points connected by line segments for instance. Thus, the equation (2.3) has to be solved on this set of points to obtain a new position and the result is generally accurate provided that the interface elements remains not too much distorted and that the connectivity does not change (i.e. no merging or pinching apart of the interface). Unfortunately, these conditions are rarely fulfilled and a simple velocity field can deteriorate rapidly the accuracy involving large distortions if the interface discretization is not modified to maintain an acceptable aspect ratio of the elements.

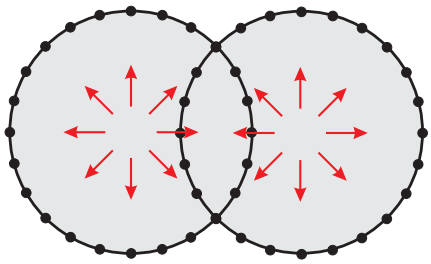


Figure 2.7: Overlapping of two growing circles

Moreover, if the connectivity of these points is modified between two consecutive time steps, the detection of the external contour can become troublesome. The illustration of this problem

is depicted in Figure 2.7 where two initial disjoint circular interfaces grow so much that they overlap. In 3 dimensions, the segments would be replaced by triangles and/or quadrangles and it is easy to conceive that it becomes quite difficult. With an implicit definition in an Eulerian framework as in the Level Set method, the extension to 3D interface is much simpler and does not introduce additional difficulties.

## 2.3 Fast Marching Method

When the speed has a constant sign, i.e for steadily propagating fronts, the interface is always moving forward or backward. Therefore, the evolution of the Level Set can be seen as a stationary problem because the interface cross a fixed point only once. The equation of movement can be modified into a stationary one to take the form of the Eikonal equation:

$$\|\nabla\phi(\mathbf{x})\| = \frac{1}{\boldsymbol{F}} \tag{2.4}$$

Hence, the initial value problem of (2.3) is now turned into a boundary value problem. This form presents the advantage that it can be solved with a very efficient numerical scheme known as the Fast Marching Method (FMM) (the most efficient technique can solve equation (2.4) in $O(N)$, where $N$ is the number of points [182]). In this case, the initial position of the interface is the boundary value of this equation and the solution $\phi(\mathbf{x})$ is a stationary function.

In general, the function $\phi(\mathbf{x})$ values do not represent the distance to the interface but corresponds to the time needed by the interface $\Gamma(t)$ to reach the point $\mathbf{x}$. Therefore, the initial interface position corresponds to the zero level of the $\phi(\mathbf{x})$ function while the intersection of the $\phi(\mathbf{x})$ with a plane $z = \tau$ will give the interface shape and position at a given time $\tau$. When solved with a constant and uniform speed $\boldsymbol{F}$=1 on the interface, the FMM corresponds also to the minimum distance from a point $\mathbf{x}$ to the interface $\Gamma$. Consequently, the Fast Marching Method can be used to efficiently compute the minimum distance to an interface $\Gamma$ that can be used to define a distance function $\phi(\mathbf{x})$ to $\Gamma$. This method has been successfully applied to a variety of problems such as image segmentation, optimal path planning, migration and seismic problems (see [154] for more details). Moreover, the FMM has also been applied in crack growth with X-FEM by Sukumar *et al.* [165].

In practice, during the evolution of the function $\phi(\mathbf{x})$ with the Level Set Method, difficulties can occur because the Level Set function may develop flat or steep gradient. Steep regions suffer from a poor precision on the gradient of the function $\phi(\mathbf{x})$ while flat regions exhibit poor accuracy on the $\phi(\mathbf{x})$ values. Thus, when solving the equation (2.3), a reinitialization step is introduced to rebuild the Level Set as a signed distance function to avoid the loss of accuracy. While it can be done by recomputing a new signed distance function to the interface, another approach consists in using the Fast Marching Method (forward and backward) to solve the Eikonal equation $\|\nabla\phi(\mathbf{x})\| = 1$ to obtain a new signed distance function $\phi(\mathbf{x})$.

## 2.4 Geometric tools

Much geometrical information of the interface can be retrieved easily from the implicit representation using the $\phi$ function values. For instance, the gradient $\nabla\phi$ is perpendicular to the isocontours of $\phi$ function and it points out in the direction of increasing $\phi$ values.

### *Normal*

Thus, the unit outward normal $\boldsymbol{N}$ to the interface is easily given by: $\boldsymbol{N} = \frac{\nabla\phi}{\|\nabla\phi\|}$.

### *Curvature*

Similarly, one can show that the curvature $\kappa$ of the $\phi$ function is given by: $\kappa = \nabla \cdot \boldsymbol{N}$. Generally, it is more convenient and accurate to compute the curvature directly from the $\phi$ values with:

$$\kappa = \nabla \cdot \frac{\nabla\phi}{|\nabla\phi|} = \quad (\phi_x^2\phi_{yy} + \phi_x^2\phi_{zz} + \phi_y^2\phi_{zz} + \phi_y^2\phi_{xx} + \phi_z^2\phi_{xx} + \phi_z^2\phi_{yy} - 2\phi_x\phi_y\phi_{xy} \\ -2\phi_x\phi_z\phi_{xz} - 2\phi_y\phi_z\phi_{yz})/\left\|\nabla\phi\right\|^3$$

so that for convex regions, the curvature is $\kappa < 0$ everywhere whereas $\kappa > 0$ for concave regions, and $\kappa = 0$ for planar regions.

### *Volume and volume integral*

If we define a Heaviside function $H(\phi)$ such that:

$$H(\phi(\mathbf{x})) \;=\; \begin{cases} 1 & \text{if } \phi \leq 0 \\ 0 & \text{if } \phi > 0 \end{cases}$$

where the sign of $\phi$ depends on the point of evaluation $\mathbf{x}$. The volume integral of a function $f(\mathbf{x})$ over the positive or negative part of the Level Set is given by:

$$\int_\Omega f(\mathbf{x})H(\phi(\mathbf{x}))d\Omega \;\text{ or }\; \int_\Omega f(\mathbf{x})(1 - H(\phi(\mathbf{x})))d\Omega$$

Using $f(\mathbf{x}) = 1$ in $\Omega$, one can calculate the volume (area) of the positive or negative part.

### *Boundary integral*

Similarly, the integration of $f(\mathbf{x})$ along the interface $\phi$=0 is given by:

$$\int_\Omega f(\mathbf{x})\hat{\delta}(\mathbf{x})\left\|\nabla\phi(\mathbf{x})\right\| d\Omega$$

where $\delta(\mathbf{x})$ is the *Dirac function*. If the function $f(\mathbf{x})$=1 on the domain $\Omega$, the surface integral corresponds to length of the interface in 2D and the area of the interface in 3D.

### *Closest interface point*

If $\phi(\mathbf{x})$ is defined as a signed distance function, the closest point on the interface $\mathbf{x}_\Gamma$ to a point $\mathbf{x}$ in the domain $\Omega$ is easily given by: $\mathbf{x}_\Gamma = \mathbf{x} - \phi(\mathbf{x})\boldsymbol{N}$.

### *Point in polygon*

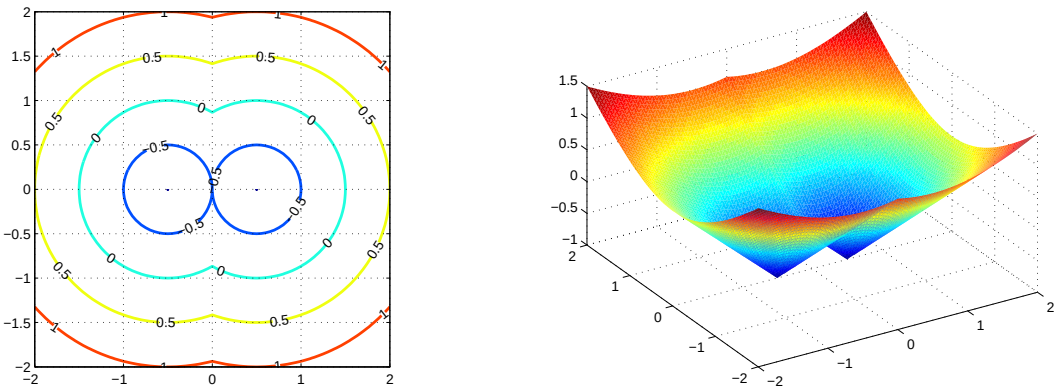
Moreover, the implicit geometrical representation allows to easily determine whether a point $\mathbf{x}$ is inside or outside the interface just by inspecting the sign of the function $\phi$. Here again, the implicit formulation of the interface is of a great help. To realize the same test with an explicit geometry, one generally uses a *Point in polygon* algorithm [168], which consists in casting a ray from the point of interest $\mathbf{x}$ to another point located far from the interface and known as outside.

Then, if the ray crosses an even number of times the interface, the point $\mathbf{x}$ is outside, while if the ray intersects the interface an odd number of times and is located inside the interface. It is obviously simpler to determine the sign of the $\phi$ function than using the aforementioned algorithm.

### *Boolean operations*

The possibility to combine different Level Sets is also one of the appealing characteristics of this method. This property allows an easy treatment of merging interfaces and connectivity modifications.

*Union*
For example, the Figure (2.8) represents the union of two Level Sets $\phi_1$ and $\phi_2$. It is built by taking the minimum of the two Level Sets (one per geometrical entity) which are computed separately on the whole grid and it represents the union $\phi$ of the two interior regions of $\phi_1$ and $\phi_2$: $\phi = \min(\phi_1, \phi_2)$.



(a) Isocontour of the union of two Level Set function

(b) Union of two Level Set function

Figure 2.8: Level Set representation of the union of two circles

*Intersection*
Similarly, the intersection of the interior region of two implicit functions $\phi_1$ and $\phi_2$ is given by the max operator.

*Difference*
The difference $\phi_1 - \phi_2$, defined as $\max(\phi_1, -\phi_2)$, corresponds to the interior region of $\phi_1$ function minus the interior region of $\phi_2$ function.

*Complement*
Finally, the complement of the interior domain of a $\phi$ function is naturally given by the opposite $-\phi$ as summarized in the Table 2.1.

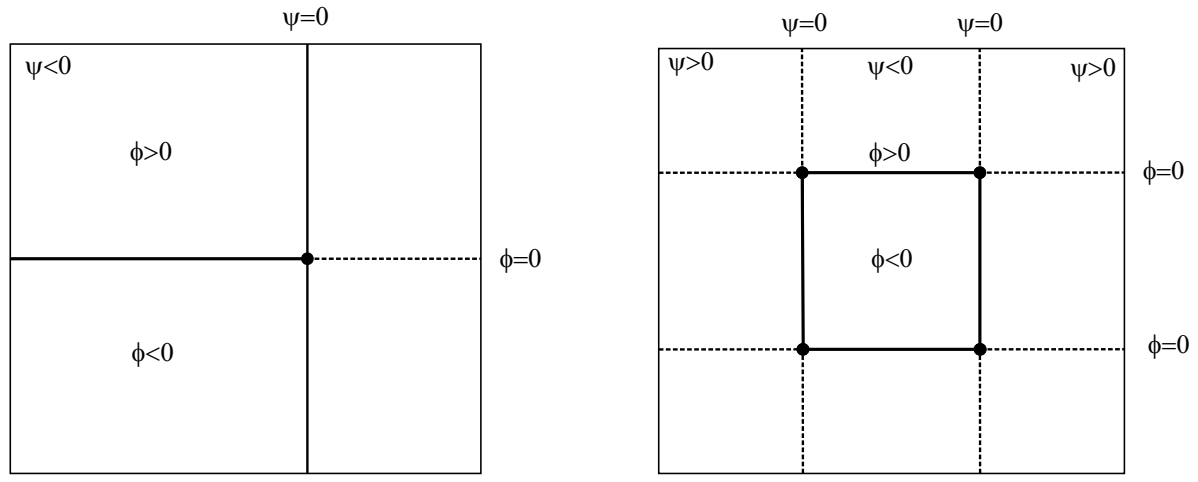
The interesting advantage of these boolean operators is that any modification of the interface topology is handled naturally in a systematic manner. Hence, if the two circles in the Figure 2.8

| Union | $\phi_1 \cup \phi_2$ | $\min(\phi_1, \phi_2)$ |
|---|---|---|
| Intersection | $\phi_1 \cap \phi_2$ | $\max(\phi_1, \phi_2)$ |
| Difference | $\phi_1 - \phi_2$ | $\max(\phi_1, -\phi_2)$ |
| Complement | $\phi^c$ | $-\phi$ |

Table 2.1: Boolean Level Set operators

reduce their radius to reach a configuration where the two circles do not overlap anymore and separate to form two disjoint curves, the interface is always defined without any technical difficulties as the set of points where the $\phi$ function is equal to zero. In fact, we can easily see from the Figure 2.8 (a) that, translating the $\phi$ function along the $z$ axis produces different interface topologies. In this case, using an explicit geometrical definition of the interfaces would have called for a modification of the topological definition of the model when the two circles overlaps.

Above, we have limited the presentation of the Level Set to closed interfaces. While there is no difficulty if the interface is open with both ends located outside of the computational domain, the representation of an open interface with both ends within the domain is not straightforward. An open interface cannot be modeled with only one interface as an open curve does not separate a domain into two distinct regions. The solution to this problem has been first proposed by Smereka [159] for spiral crystal growth. It consists in introducing a second Level Set function $\psi$ to bound the $\phi$ function inside the domain. Thus, the interface is still defined as the zero isocontour of the function $\phi$ but is bounded to the region where the second function $\psi$ is $\leq 0$ or $\geq 0$ as depicted in Fig. 2.9.




(a) Level Set representation of a semi-infinite interface line

(b) Resultant interface on the mesh

Figure 2.9: Level Set representation of opened and polygonal closed interface

For Figure 2.9 left, we need one Level Set function $\phi$ to define an infinite interface and this interface is bounded with the second Level Set function $\psi$. The interface is thus defined as the set of points where $\phi = 0$ and $\psi \leq 0$. For the Figure 2.9 (a), if the two functions $\psi$ and $\phi$ are constructed such they intersect twice the plane $z = 0$ (a ridge function or a paraboloid function for instance), a squared region can be defined as the region where $\psi$ and $\phi$ are both $\leq 0$ i.e.

or $\max(\phi, \psi)$, requiring only two functions. However, in the most general case, a square or a $n$-side polygon is defined with four or $n$ Level Set functions, each of them defining one side of the polygon. This technique using two distinct Level Set functions is generally used to represent a crack and the crack tip inside a computational domain [162]. Remark also that in this precise case, the interface does not partition the computational domain in two clearly distinct parts (inside and outside). Using a Level Set description, it is possible to partition a region into a maximum of $2^n$ parts by using $n$ Level Sets.

## 2.5 Numerical treatment of Level Sets

### 2.5.1 Construction of signed distance functions

When one has to implement a representation of interfaces in practice, neither the explicit nor the implicit representation stores the exact position of the points on the interface. Instead, we compute the exact position of a set of points on the curve or on the surface and the remaining points are interpolated.

Hence, with an explicit representation, the interfaces, curves in 2D or surfaces in 3D, are approximated, i.e. discretized into a finite set of points. Depending on the accuracy required, a curve $c = \mathbf{x}(s)$ defined on an interval $[s_0,s_n]$ is then divided into different intervals $s_0 < s_1 < s_2 < s_3 < \ldots < s_n$ which are not necessarily equispaced in the physical space. For each values of $s_i$, the corresponding position of the point is stored into an vector. Obviously, as the number of intervals is increased the resolution of the curve is improved. Between these points, the curve is interpolated to determine the position of any additional point lying in the interval $[s_i, s_{i+1}]$. In practice, the most simple interpolation scheme consists in representing the interval $[s_i, s_{i+1}]$ as a line segment or with a spline interpolation.

With an implicit representation, it is not straightforward to determine *a priori* the positions of a finite set of points on the interface. Instead, we know the values of the implicit function $\phi$ on a set of points and an interpolation scheme is needed to obtain the values on the other points. Generally, when using the level set method, the computational domain is discretized on a uniform Cartesian grid to ease the mesh generation and the computation of geometric quantities such as derivatives or curvature. Then, the value of the Level Set function is computed at each grid point and interpolated on the whole grid. Different types of interpolation can be used, for instance a Lagrange Polynomial interpolation [186] (which is the most common method) or a Radial Basis Function interpolation [41]. To obtain the position of the interface, the isocontour $\phi = 0$ requires to be interpolated from the grid points values, which is a rather standard procedure that can be realized by contouring algorithm. With an explicit representation we know the exact position of some points on the interface (the sampled points). But with an implicit formulation it is possible that we do not know the position of any point exactly. In fact, if no grid point coincide exactly to a point on the interface, no grid point supports a $\phi$ value equal to 0. Therefore all the points of the interface are interpolated and it may happen that no point is exact.

In practice, as an interface represented by an analytical expression is too restrictive to model complex geometries, the Level Set $\phi$ is usually constructed as the signed distance to the boundary $\Gamma$ and discretized on a fixed mesh. The sign of this function is then different on each side of the interface $\Gamma$ and is determined with the normal $\mathbf{n}(\mathbf{x_\Gamma})$ to the interface or with a simple counter-clock-wise method when marching along the interface. Thus, if the normal $\mathbf{n}$ can be obtained

from the interface description, we can express the signed distance function as:

$$\phi(\mathbf{x},t) = sign(\mathbf{n}\cdot(\mathbf{x}-\mathbf{x_\Gamma}))\cdot\min_{\mathbf{x_\Gamma}\in\Gamma}\|\mathbf{x}-\mathbf{x_\Gamma}\| \quad \forall\mathbf{x}\in\Omega$$

where $\mathbf{x}_\Gamma$ is the closest point from $\mathbf{x}$ to the interface (i.e. the projection of $\mathbf{x}$ on $\Gamma$). After discretization on a grid, the points $\mathbf{x}$ are the nodes of the grid. If the Level Set description is used to represent a closed geometrical object such as a mechanical structure, the iso-zero contour corresponds to the boundary of the structure. The sign is generally chosen to be negative if $\mathbf{x}$ is inside and positive outside the interface (see Fig. 2.13). When considering the inside as the negative, the normal to the interface computed with $\boldsymbol{N} = \frac{\nabla\phi}{\|\nabla\phi\|}$ is automatically oriented outward which is consistent with a geometrical CAD representation where the surface surrounding a 3D object is usually oriented outward.

When the Level Set is constructed from a NURBS curve, the Level Set can be computed by dividing the curve into small segments and by computing the distance between the mesh nodes and these segments. However, to limit the number of operations and the computational time, each segment is stored into an octree structure (using ANN library [120]) in order to compute the distance from a given mesh node to the closest points on the interface. For NURBS surfaces, the same procedure is applied by subdividing the surface into patches.

### 2.5.2 Interpolation of iso-zero Level Set on an edge

When the Level Set is described with an analytical function, the generation of the corresponding Level Set is straightforward as in the example of the unit circle. However, as the Level Set $\phi$ can be far from a signed distance function, the determination of the iso-zero from the interpolated Level Set $\phi^h$ can suffers from inaccuracies as the slope of the Level Set is not constant anymore. As mentioned in section 2.5.3, the Level Set function $\phi$ is interpolated with classical finite element shape functions. Hence, along an element edge the Level Set function is interpolated from the nodal Level Set values $\phi_i^h$.

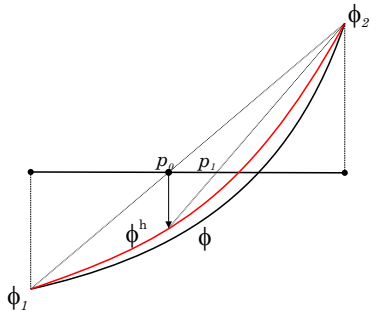



Figure 2.10: Interpolation of the Level Set and computation of $\Gamma$ with secant method

As depicted in the Figure 2.10, an estimation of the iso-zero Level Set position $p$ is obtained using the secant method [117] corresponding to a linesearch with an incertitude interval:

$$\begin{aligned} p &= p_1 + t(p_2 - p_1) \\ t &= \frac{-\phi^h(p_1)}{\phi^h(p_2) - \phi^h(p_1)} \end{aligned}$$

where $\phi^h(p_1)$ and $\phi^h(p_2)$ are the Level Set value at node 1 and 2 respectively. The position of $p$ is always interpolated linearly independently of the shape function order used to interpolate $\phi$. From this first estimation of the iso-zero position $p$, the corresponding Level Set value is obtained $\phi^h(p)$ and the procedure is repeated until $\left|\phi^h(p)\right| < \epsilon$ by replacing $\phi(p_1)$ by $\phi^h(p)$ if $\phi(p_2)\phi^h(p)<0$ (or $\phi(p_2)$ by $\phi^h(p)$ if $\phi(p_1)\phi^h(p)<0$). Moreover, the evaluation of $\phi^h(p)$ is obtained from interpolation of the discretized function $\phi^h$ and no more call to the Level Set function $\phi$ should be realized unless problems can be encountered when evaluating the sign of the Level Set.

### 2.5.3 Interpolation of the Level Set on a FEM grid

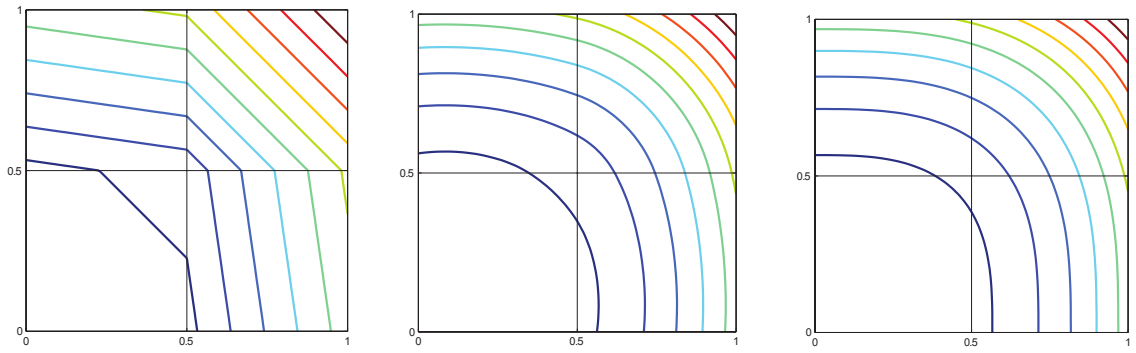
Inside the elements, the Level Set function is generally approximated on the mesh using the finite elements shape functions as:

$$\phi^h(\mathbf{x}) = \sum N_i(\mathbf{x})\phi_i \tag{2.5}$$

where $N_i$ are the FEM shape functions and $\phi_i$ the nodal values of the Level Set at node $i$. Hence, the Level Set values are only known exactly at the nodes and interpolated elsewhere in the mesh. As a consequence, the accuracy of the Level Set approximation depends on the order of the shape function used in (2.5) for $N_i$. The figures 2.11 illustrates the interpolation of a Level Set function $\phi$ of order 3 defined as:

$$\phi(\mathbf{x}) = y^3 + x^3 - \frac{1}{2}$$

with 3 different orders of approximation (bilinear, biquadratic and bicubic) on a mesh composed of 4 quadrangular elements.

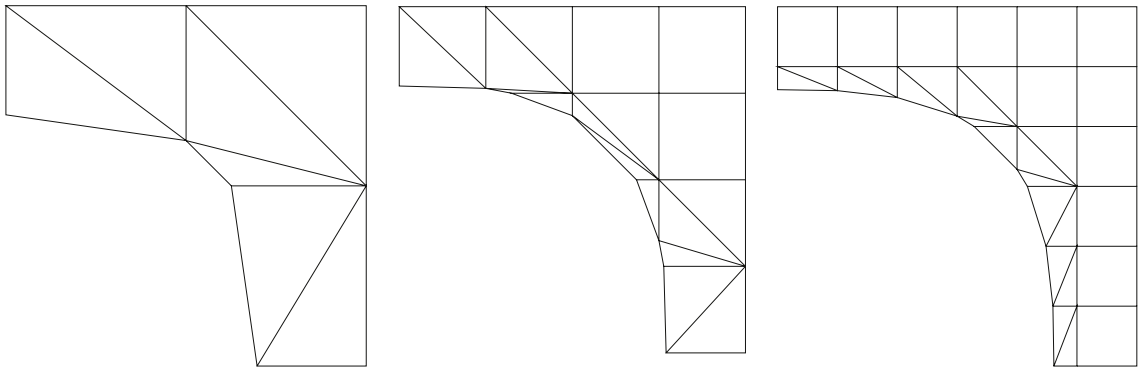



(a) Order 1 (b) Order 2 (c) Order 3

Figure 2.11: Interpolation of the Level Set on 4 elements

The Figure 2.12 illustrates the sliced mesh obtained with the corresponding mesh.

To obtain a better geometrical accuracy and a better Level Set representation while keeping a low order FEM shape function during problem resolution, one can of course use a non conforming adaptive mesh refinement, a quadtree/octree submesh, or simply define new nodes on FEM elements to define use different approximation orders for the geometrical representation and physical fields.

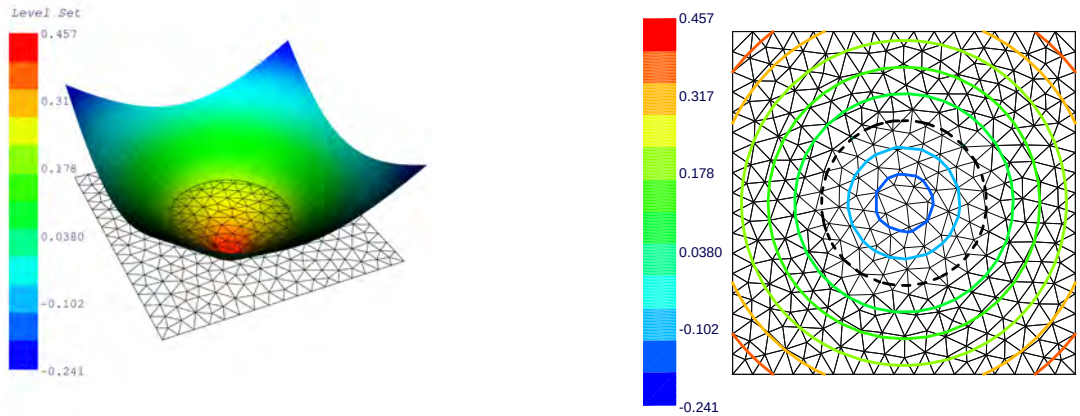


(a) Degree 1 (b) Degree 2 (c) Degree 3

Figure 2.12: Mesh slicing with 3 different approximation orders

### 2.5.4 Geometrical accuracy

From equation (2.5), one can notice that with linear shape functions on a triangular mesh, the interface is linear inside the elements and the interpolation of the Level Set is piecewise linear on the mesh. In Figure 2.13, one can see that a rather coarse mesh can already gives a good resolution / representation of a unit circle using a linear interpolation within each element.



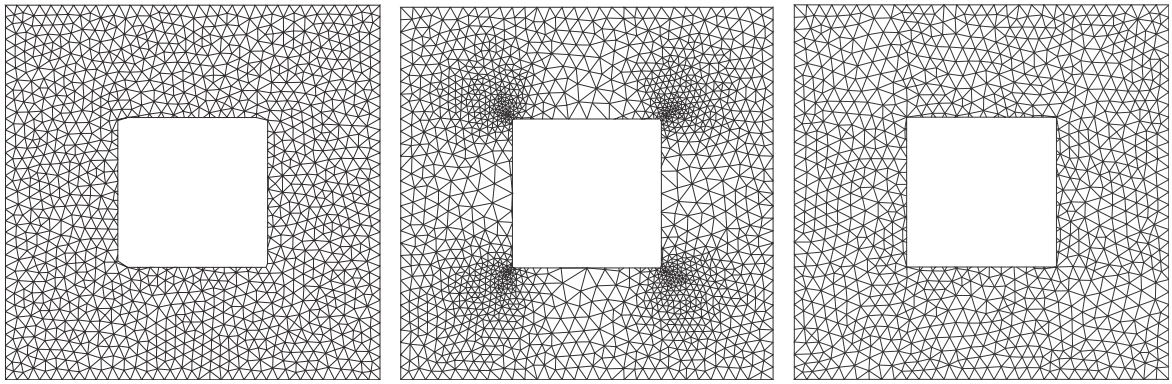

(a) Representation of Level Set function

(b) Isocontour of the Level Set function and interface in black dashed

Figure 2.13: Geometric representation of a circular interface on a first degree triangular mesh

However, the resolution is not always so good. For instance the Level Set description fails at representing accurately complex geometries containing corners or small details compared to the mesh size. The Level Set description tends to erase or smooth out these details when using linear interpolation inside an element as depicted in Fig. 2.14.

The Figure 2.14 represents a square interface represented by an implicit function discretized on a triangular mesh. In Figure 2.14 (a), the representation is not able to represent sharp corners as they are smeared out. The reason is that the Level Set is not able to capture a corner in the most

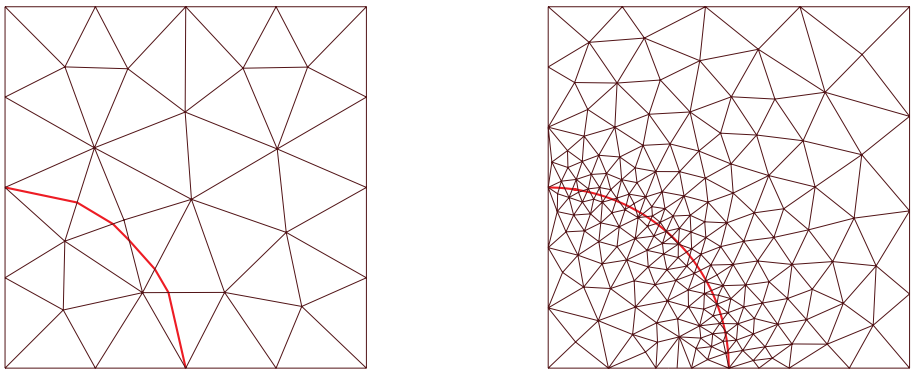


(a) Standard mesh (b) Adapted mesh (c) Points exactly located at corners

Figure 2.14: Geometric representation of corners

general case because it can not represent a kink inside an element but only a line, when using a linear interpolation, or a curve, with higher order approximation. Figure 2.14 (b) illustrates that when the mesh gets more refined near the corners, the representation tends to be more accurate but still fails at representing the corners. Finally, Figure 2.14 (c) illustrates that it is possible to obtain an accurate representation of a corner without any refinement if a grid point is placed exactly at the corner position.

The quality of the representation depends on the curve characteristics and on the mesh densities. Therefore, when the boundary is made of non smooth curves, the quality of the mesh is very important to obtain a satisfying geometrical representation of the model. Both the geometrical and numerical errors from the model are reduced as the mesh gets refined. As example, we illustrate the effect of the mesh refinement for the representation of a quarter circle on a plate (Fig. 2.15).



(a) Coarse mesh (b) Locally adapted mesh size map

Figure 2.15: Effect of the mesh refinement on the quality of the Level Set representation

The Figure 2.15 (a) presents a rather coarse mesh and leads to a highly kinked geometrical boundary whereas the Figure 2.15 (b) has a more accurate geometrical representation of the

circle. The refinement is only needed for the elements located close to the boundary and, the mesh conformity is not necessary. To obtain the Figure 2.15 (b), we have adapted the characteristic mesh length of the mesher only in the vicinity of the iso-zero Level Set. The quality of the representation depends on the curve characteristics and on the mesh density. Both the geometrical and numerical errors from the model are reduced as the mesh gets refined.

When the Level Set is defined analytically, as in Figure 2.15, the definition of the map of the characteristic mesh length is easy to obtain and can be defined without an initial mesh. However, in practice, we do not know the position or the aspect of the Level Set and the mesh refinement procedure should be iterative. Thus, a first initial coarse mesh should be created in order to be able to interpolate the Level Set and then compute its geometric quantities to determine the regions where the element densities should be increased and where the mesh should be refined.

From Figure 2.13, we can clearly see that the geometric quantity to take into account is the radius of curvature. As the interface is generally considered as a straight line or plane inside the elements, the size of the elements should be sufficiently small to obtain a given geometrical accuracy. In case of nearly linear or weakly curved interface no mesh refinement is needed. However, a global mesh refinement is wasteful as only a few elements located near the zero level set interface need to be refined. Therefore, in addition to the curvature criteria, it is worthwhile to limit the refinement in the vicinity of the elements that are close to the interface.

In Ref. [113], Moës *et al.* have presented a simple criterion to adapt the mesh. The strategy consists in reducing the characteristic element size $l_c$ at the proximity of the iso-zero Level Set where the radius of curvature $\rho$ of the interface is small until the following conditions are stratified:

$$l_c > \alpha\rho \quad \left|\bar{\phi}\right| < \beta l_{tot} \tag{2.6}$$

$\left|\bar{\phi}\right|$ is the level set value at the element barycenter, $l_{tot}$ is the side length of the basic cell and the dimensionless parameters $\alpha$ and $\beta$ govern the extent and the level of refinement. For elements presenting a $l_c$ value higher than the radius of curvature of the Level Set and a centroidal Level Set value smaller than a given threshold, the elements are tagged for subdivision. Starting from an initial coarse mesh, several refinements are usually required to meet the condition (2.6). Finally, as meshes get more refined, the evaluation of the curvature $\rho$ becomes more accurate and the process ends when no element is tagged.

## 2.6 A Level Set based Constructive Geometry

The constructive solid geometry (CSG) is a technique widely used in solid modeling and Computer Aided Design (CAD) [60]. The idea of this technique is to combine simple objects called primitives in order to define complex geometries. Typically, the primitives have very simple shapes such as cuboids, cylinders, prisms or spheres but more complex geometries such as curved objects can be considered too. Each combination of primitives is obtained by using Boolean operators: union, intersection, difference (see section 2.4). These operators generally act upon two objects and produce a single compound object. The union of two objects results in an object that encloses the space occupied by the two primitives. Intersection results in an object that encloses the shared space of the two given objects. Then, the difference results in the first object minus the space where the second intersected the first (see Fig. 2.16).

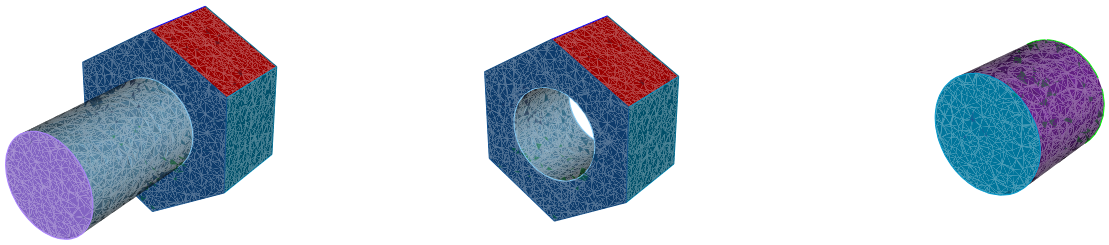
(a) Union of an extruded hexagon and cylinder

(b) Difference of an extruded hexagon and cylinder

(c) Intersection between extruded hexagon and cylinder

Figure 2.16: Boolean operations in constructive solid geometry

In constructive solid geometry, the primitives always divide the space in two distinct parts and the related Boolean operators are used to combine or extract different space regions surrounded by the primitives. Indeed, one can easily see that the Level Set function also splits a domain in two parts with respect to its sign value and that the Level Set operators act just as the Boolean CSG operators. Therefore, these operators can be used together with several Level Set primitives to represent complex geometries.



(a) Union

(b) Difference

(c) Intersection

Figure 2.17: Example of a CSG Level Set Model

Practically, following the approach of CSG, a solid structural geometry can be procedurally modeled as a compound of different Level Sets and organized as a binary tree where the leaves are Level Set primitives (see Fig. 2.18).

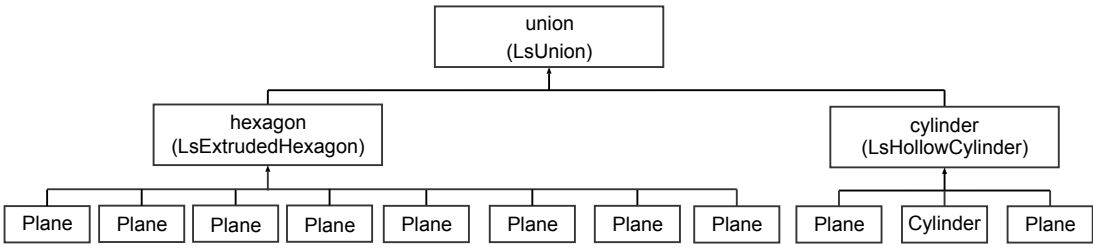



Figure 2.18: Tree representation of the Level Set Model for model of Figure 2.17 (a)

## 2.7 Level Set and the eXtended Finite Element Method

Applied with the eXtended Finite Element Method, the Level Set is defined on the mesh used for the structural analysis and a geometrical degree of freedom representing its Level Set function value is associated at each element node. The construction of this Level Set can be made from an analytical description in simple cases or from NURBS curves and surfaces or, in a general case, from a given set of points as described in Ref. [25]. The discrete Level Set is then interpolated on the whole design domain with the classical shape functions of the finite element approximation.

As the Extended Finite Element Method will be introduced in the next chapter, we will not go into the details about the utilization of the Level Set with X-FEM. However, we can already mention some interesting features of coupling these two methods. The basic idea of the X-FEM method is to add special shape functions to some nodes in order to model a discontinuous behavior related to the interface. Therefore, the first difficulty arises when we have to find the elements which support these additional shape functions. For instance, in a material-void interface, only the elements that intersect the boundary of the hole require a special treatment. Therefore, with an explicit geometrical representation, one would have to deal with complex geometrical considerations in order to find these partially filled elements. With a Level Set representation, these elements are easily detected as they have at least one nodal negative Level Set value. In the X-FEM, several additional shape functions present a change of sign on each side of the interface. The sign of these functions can be directly given by the sign of the Level Set. In fracture mechanics, modeling the dynamic growth of arbitrary cracks in 3D remains a challenging problem. To this end, several authors [162, 165] coupled the X-FEM with the Level Set method in order to benefit from the efficient geometrical representation of the Level Set description which removes the need to represent and to maintain the geometry of the crack during its evolution. Furthermore, this geometrical representation can rely on efficient algorithms allowing a free evolution of the crack.

## 2.8 Implementation of the Level Set representation in OOFELIE

In this thesis, the Level Set is primarily used as a geometric representation to describe the inner or outer boundaries of the structure. To this end, we have introduced a new Level Set package inside the C++ OOFELIE software in order to represent the geometry of interfaces with the aim of combining it with the X-FEM elements (see chapters 3 and 4). Within this Level Set framework, one is now able to build several basic shapes defined either with analytical functions or simple geometric primitives such as:

- quadric functions : $A\mathbf{x}^2 + B\mathbf{x} + C$
- circles, ellipses, generalized super ellipses ...
- planes, rectangles, polygons ...

Then, the capability of inserting several Level Sets is also available and all these basic shapes can be combined together into more complex primitives by using the different Level Set boolean operators (union, intersection and difference). Moreover, as general geometries can not always be described from these basic shapes, we have also introduced the possibility to build a Level Set from a set of points, a NURBS curve/surface and from a CAD model through STEP, IGES and STL exchange format files.

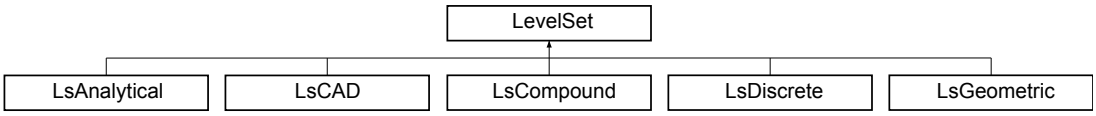



Figure 2.19: Representation of the LevelSet classes inheritance

From computer implementation point of view, the Level Set module is made of a base class LevelSet that serves as an interface through a model described as a Level Set. Hence, one can obtain local values, interpolated values, curvature, etc and for optimization purposes, the values and the number of parameters describing each Level Set. Then four classes inherit from the Level Set base class and correspond to a classification by nature of Level Set (see Fig. 2.19):

- The analytical Level Set class gather all implicit geometries that can be modeled with an analytical definition. In practice, this class is specialized into a general quadric: $A\mathbf{x}^2 + B\mathbf{x} + C$ Level Set class that can be tuned by the user with parameters $A, B$ and $C$ and a set of predefined quadric such as ellipse or circle.
- The geometrical subset classes encompass all simple geometric primitives such as planes, circles for instance.
- Discrete class is used for creating a Level Set from a scalar field defined on mesh grid.
- The CAD Level Set class represents a Level Set coming from a CAD model file.
- The Compound Level Set class represents predefined complex Level Set and Level Set operators (union, intersection, difference). See Fig.2.20.

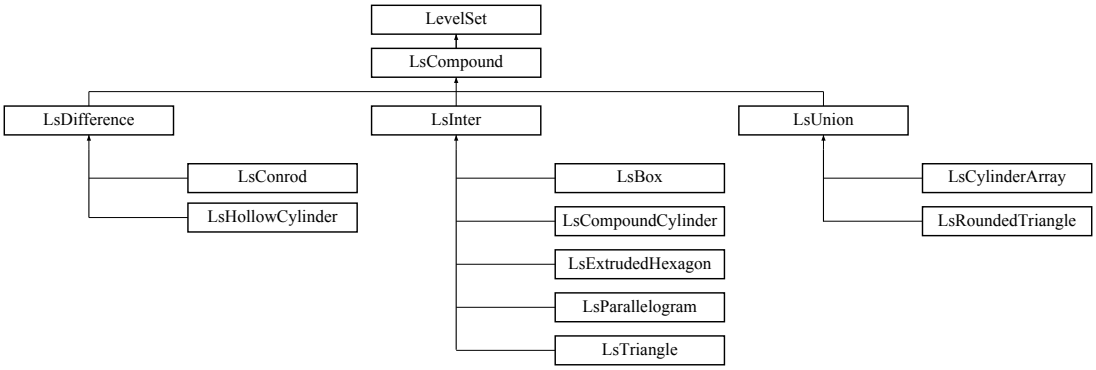



Figure 2.20: Representation of the LsCompound classes inheritance

The three boolean operators inherit from the compound Level Set class. They are considered as a Level Set and not really as an operator. Inside each of this compound Level Set, one can pile up a series of others Level Set and the output is the union, intersection or difference computed on the pile. Hence, they are not true binary operators as the union operation can be proceeded on more than two Level Sets. Again, this class is specialized for different assemblies such as a hollow cylinder, a connecting rod (or conrod) or a box. It is this compound class that is use to build a Level Set CSG model. Finally, one can pile all Level Set objects into a tree structure in order to build a complex CSG Level Set model. Finally first order method to solve the Hamilton-Jacobi has been implemented.

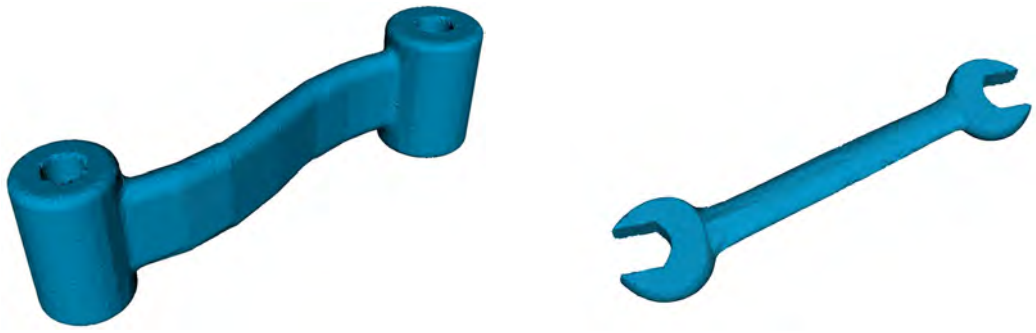
### 2.8.1 Constructing Level Set geometries from CAD models

The geometric CAD approach has now become a standard in Computer Aided Engineering (CAE) [109]. Hence, every FEM model generally relies on a geometry coming from a solid Modeler. As we pointed out in the introduction, explicit geometric representation is one source of the poor performance of classical shape optimization. And it is for this reason, among others, that we prefer to model the geometry with a Level Set Description. The CSG Level set approach is very versatile and, without considering meshing limitation, any geometry starting from a cylinder to a complex industrial body can be modeled using a CSG Level Set tree. However, the main drawback of building a CSG Level set is that there is no existing commercial tool allowing the construction of such complex geometry with Level Sets. Moreover, it would be a grueling process to create such a tool and to replace the standard CAD modelers that are well-established in engineering. It is more effective and practical to design a tool able to convert a CAD representation into a Level Set one.

During the project EFCONIVO, we identified three methods that could be applied to obtain a Level Set from a real geometry modeled with a CAD engine. If one has access to the so-called *build tree* i.e. the successive steps of operations (union, intersection, extrusion ...) that have conducted to the final geometry, one can imagine to translate each geometric entity into its corresponding Level Set and to apply the successive operations realized within the CAD modeler to the Level Set entities. This technique is the most simple as one has "only" to translate each geometric entity and boolean operators to their Level Set counterparts and *replay* the successive operations stored in the *build tree* to the Level Set entities. Unfortunately, this approach is difficult in practice because it requires to have access to the *build tree*, which is generally not possible. Most known CAD engines are unfortunately proprietary solutions such that it is not possible to read their file format, to get access to the *build tree* or to fetch the parameters of the shapes such as the radius of a circle. Moreover, it might happen that the *build tree* is even not stored. With standard CAD output files such as STEP, IGES, STL, the problem is different as the geometry is exported as a non-editable block and the *build tree* is not exported into this file format.

As all geometric modelers usually offer the capability to obtain STL triangularization of geometric surfaces, another strategy that has been proposed and implemented into Gmsh [75] by Geuzaine and Remacle is to compute the signed distance to the geometry. The principle of this algorithm is to discretize the geometry described in IGES or STEP format with a STL triangulation and to compute the distance from the mesh nodes to these discretized surfaces. In order to obtain a reasonable computational time, it is preferable to limit the meshing domain to a bounding box around the geometry and to use a hexahedral mesh that is simple and fast to generate. Moreover, to limit the number of distances to be computed, one can take advantage of the regular topology relation of hexahedral mesh to select only a set of elements close to the surface to compute the minimal distances.

Generally, this approach gives good results as illustrated in Fig. 2.21. However, it is actually only well suited for structures that present high aspect ratios. With small thicknesses, the generation of the active elements selects all elements that are inside the body and the number of elements across the thickness remains small. However, with structures that do not exhibit a small thickness, the strategy to select the surrounding element generates only the Level Set values around the skin of the structure. Therefore, in order to assign a sign to each node of the structure, the Level Set value has to be extended outside 1/whether using a neighborhood algorithm that could



(a) Conrod (b) Wrench

Figure 2.21: Representation of a geometry realized from a CAD file to STL

assign a sign from know values or 2/by computing all distances which may be computationally expensive. Unfortunately, a tool based on the neighborhood has not been yet implemented inside Oofelie or Gmsh to this end and we have thus chosen to compute the Level Set on all nodes. The same remark can be stated if one uses unstructured tetrahedral mesh topology because the definition of active neighboring elements is quite difficult. If the mesh is composed of elements of uniform size, one can select a radius based on this average element size to determine a neighboring region. However, in practice, the size of the tetrahedral elements is far from being uniform. The definition of a neighborhood radius is more difficult to carry out and it can deteriorate the performance of the algorithm 1.

Also, as we only store the minimal distance from all geometric surfaces, the Level Set representation is only made of one Level Set and it is not possible to represent exactly sharp edges. This problem is inherent to the Level Set representation that performs well with smooth geometry but not with complex boundaries as presented in section 2.5.4. Again, a higher accuracy can be obtained for the representation if the mesh is refined in the vicinity of these sharp edges. Conforming refinement or non-conforming octree mesh based refinement can also be considered.

Exact representation can be obtained if one Level Set is assigned to each geometrical surfaces. In this case, one has a set of Level Sets and boolean operators can be used to combine objects in order to construct the geometry. Unfortunately, this approach may be difficult in practice as it also needs the *build tree* to determine which operator has to be used between the Level Set entities. Furthermore, the *build tree* is not sufficient to obtain an accurate representation of sharp corners and edges because the STL surface generated are finite surfaces. Therefore, if a corner composed of 2 surfaces lies inside an element, each surface does not divide the element into two distinct parts. So, it is not possible to represent sharp corners and one should better take into account of the topology information contained in the STEP file format in order to realize the Level Set from a CAD model.

## 2.9 Other methods for constructing a Level Set

### 2.9.1 Level Set construction with Radial Basis Function interpolation

Belytschko *et al.* [25] have proposed to use a function interpolation technique introduced in the late 90's in computer graphics for reconstructing implicit surfaces from a set of points. The idea is

to find a function $f(\mathbf{x})$ which implicitly defines a surface and satisfies to the equation $f(\mathbf{p}_i) = 0$, where $\mathbf{p}_i$ is a set of points belonging to the surface. Radial basis functions are generally chosen to approximate the function $f(\mathbf{x})$ as they have proven to be one of the most accurate and stable method to solve scattered data interpolation problems. The resulting interpolated functions becomes:

$$f(\mathbf{x}) = \sum_{i=1}^{n} \lambda_i b_i(\|\mathbf{x} - \mathbf{p}_i\|) + P(\mathbf{x}) \tag{2.7}$$

where $\lambda_i$ is the weight given by the radial basis function $b_i$ centered at point $\mathbf{p}_i$. $P$ is a first-degree polynomial added to ensure positive-definiteness of the solution. To avoid the trivial solution of a function $f(\mathbf{x}) = 0$ everywhere, it is necessary to add off-surface points with a given non-zero values. The problem consists then in finding a function $f(\mathbf{p}_i) = 0$ on $\mathbf{p}_i$ and equal to $d_j$ on $\mathbf{p}_j$, the off-surface points.

Practically, when a geometrical description is available, on and off-surface points are easily obtained. When no geometry is available, only a given set of surface points, obtained from digital scanner for instance, are known. An approximate surface normal can be obtained (see Turk and O'Brien [171]). To solve this equation for the set of weights $\lambda_i$ that satisfy the constraints $f(\mathbf{p}_i) = d_i$, each $\mathbf{p}_i$ is inserted in equation (2.7):

$$f(\mathbf{p}_i) = \sum_{j=1}^{n} \lambda_i b(\|\mathbf{p}_i - \mathbf{p}_j\|) + P(\mathbf{p}_i) = d_i$$

which can be written in matrix notation:

$$\begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1n} & p_1^x & p_1^y & p_1^z & 1 \\ b_{21} & b_{22} & \cdots & b_{2n} & p_2^x & p_2^y & p_2^z & 1 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots & \vdots \\ b_{n1} & b_{n2} & \ddots & b_{nn} & p_n^x & p_n^y & p_n^z & 1 \\ p_1^x & p_2^x & \cdots & p_n^x & 0 & 0 & 0 & 0 \\ p_1^y & p_2^y & \cdots & p_n^y & 0 & 0 & 0 & 0 \\ p_1^z & p_2^z & \cdots & p_n^z & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \\ p^x \\ p^y \\ p^z \\ 1 \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

where $b_{ij} = b(\|\mathbf{p}_i - \mathbf{p}_i\|)$ and the matrix is therefore symmetric. Initially, the radial basis functions used in computer graphics were global functions, leading to a full matrix in the system. Lately, Morse *et al.* [118] proposed locally supported radial basis functions and Ohtake [129] proposed a multi-scale approach to reduce the computational effort and to increase the accuracy locally. Actually, using a radial basis function, the surface reconstruction can be implemented to very large datasets with a relatively low computational time making this approach very appealing for generating Level Set functions.

### 2.9.2 Level Set construction with Fast Marching Method

Finally, one can also generate the Level Set function by using the Fast Marching Method. Given an initial set of computed nodal Level Set values on the nodes located closed and on both sides of the interface, the distance function can be obtained very efficiently by solving the Eikonal equation using FMM. As this method can handle only monotonous speed function, this problem is solved once on each side of the interface with a constant unit speed function.

Chapter 3

# The eXtended Finite Element Method

This chapter provides an introduction and a review of the eXtended Finite Element method. At first, the different methods that led to the development of enriching techniques are presented. Then, a simple example is used to illustrate the concept, key points and different approaches of the enrichment techniques, and the general aspects of the method are introduced. The X-FEM is then introduced for three main practical approximations used in mechanics to provide a global view of the enrichment problems and strategies.

## 3.1 Introduction

Since the beginning of the 60's, the Finite Element Method (FEM) has been very popular for finding approximate solutions to differential equation problems. Initially developed for the analysis of mechanical structures, it was identified early on as a versatile method that could be applied successfully to a wide class of physical and engineering problems. Thus, it was rapidly adopted by researchers and scientists because of its ability to deal with a variety of problems ranging from solid mechanics, electromagnetism, heat transfer, fluid dynamics ... The large enthusiasm encountered by this method has greatly contributed to its tremendous development. Moreover, thanks to the advancements of the computer science and the rapid development of digital computer hardware, the FEM has become an effective method for practical engineering problems. Consequently, due to its simplicity of implementation and its robustness, the industry has now widely adopted the FEM for solving daily problems. While reserved at the beginning to a class of applications requiring high performance such as aerospace applications, the FEM is intensively used to design a large area of products used in every day life.

For physical problems that admit smooth solutions, the standard FEM function space made of piecewise continuous polynomial functions is generally sufficient to obtain accurate results and ensure the convergence. However, due to the nature of the Finite Element approximation space, the study of problems with solution involving discontinuities remains challenging. To accurately model this class of problems, the construction of an appropriate and variable mesh is needed because the element topology has to be aligned with the discontinuity. For instance, dealing with the jumps in the response field arising with cracks, the mesh has to follow the discontinuity so that the edges or faces are coincident with the crack whereas the nodes must be placed on each side of the crack allowing material separation along the crack surface. For problems in which kinks or high gradient are likely to appear, a locally significant mesh refinement or a high

order finite element approximation is generally requested to obtain an accurate solution. Hence, one can easily notice that a special attention has to be paid to the mesh generation step. While the mesh generation technology has continuously improved the quality and the robustness of available finite element meshers, a human intervention is still needed to ensure that the mesh exhibits a good quality. This becomes a highly challenging task when the problem involves evolving discontinuities, therefore needing to repeat the mesh generation task at each time step. Following the evolution of a single discontinuity in 2D problems is affordable but this becomes very arduous with multiple discontinuities or with 3D problems. Moreover, the multiple regenerations of the mesh are generally expensive in terms of computational cost and in general results in a loss of accuracy when the projection of physical quantities between successive meshes is required. In Idelsohn *et al.* [86], the authors claims that: *It is widely acknowledged that the 3D mesh generation remains the highest part of the total man-hours devoted to solve computational mechanics problems. Also, the major problems come from automatization issue. The generation time remains unbounded, even using the most sophisticated mesh-generators. For a given geometry, an initial mesh can be obtained very quickly, but it may also need several iterations, including manual intervention, to achieve an acceptable mesh.*

From this statement, much attention has been devoted to the development of the so-called *meshless* or *meshfree* methods that try to overcome the difficulties related to the mesh. The idea of these methods is to get rid of the mesh and to define the FE approximation by constructing the approximate solution on a set of *sprinkled* points on the computational domain. This approximation is constructed on the nodes only, with associated weight functions having compact support with a simple shape, such as a circle in 2D or a sphere in 3D for instance. Over the years, this idea has attracted a lot of researchers and several meshless methods have been proposed in the literature. Among others, we can mention the Element Free Galerkin (EFG) proposed by Belytschko *et al.* [22], the Reproducing Kernel Particle Method (RKPM) [103] of Liu *et al.*

The main advantage of these methods is that the onerous mesh generation of conventional mesh based method is circumvented. They can easily cope with evolving discontinuities without remeshing and adaptive mesh refinement is easily accomplished by adding or removing nodes. While these methods have been applied successfully to a wide range of applications such as crack propagations [23], they suffer from some difficulties in practice:

- The imposition of the essential boundary conditions is not straightforward because the shape functions are not interpolants (lack of the Kronecker-$\delta$ property).
- The computational cost of meshless method is higher than FEM.
- The shape functions are not polynomial but rational functions and requires therefore careful integration techniques demanding high-order integration schemes.
- The shape functions have to be computed depending on the geometrical distribution of the nodes.
- Due to the numerous differences between FEM and meshless methods, the computer implementation differs significantly.

To avoid some inherent drawbacks of the meshless methods still benefiting from the advantage of the smoothness, hybrid methods coupling meshless and mesh-based methods have been proposed [141]. However, all the aforementioned drawbacks have severely limited the development

and the adoption of meshless methods in the industry. To the author knowledge, no commercial software has actually implemented a meshless solution.

The high degree of continuity in the solution field of the meshless method is in general a great advantage as the field derivative are smooth. Conversely, when the exact solution of the problem exhibits discontinuities, the accuracy of these methods can be rather poor. Thus the meshless methods solve the mesh related problem but face the same difficulties than the FEM with non smooth problems. To circumvent what is seen as a drawback in a non smooth problem such as crack analysis, Fleming and Belytschko [20] proposed to enrich the approximation space of EFG trial functions by including the crack tip asymptotic field into the displacement field. With this enrichment technique, Krysl [95] has been able to model crack propagation in 2D and 3D.

In 1996, Melenk and Babuska [110] showed that the classical finite element basis can be extended to represent a specific given function on the computational domain and that some advantages found in the meshless approach can be realized using a partition of unity method (PUM). The idea of PUM is to enrich or to extend the finite element polynomial approximation space by adding special shape functions, which can represent exactly an *a priori* known behavior of the solution. Initially, the aim of adding enrichment functions was to improve the performance of the classical finite element approximation on the entire domain and the enrichment was thus carried out globally (on all elements). In their seminal work on PUM, Melenk and Babuska introduced a global harmonic polynomial enrichment to deal with a globally non-smooth solution as it can be the case with high frequency solution of the Helmoltz equation. They showed that such technique of embedding additional function yields accurate solutions and that optimal rates of convergence could be obtained. The idea of enriching the space with *custom* shape functions was already known, for instance in the Global-local method [119]. However, this method did not get a lot of success due to the fact that the enrichment is global, which destroy the banded structure of the stiffness matrix. In the PUM, the central idea is to multiply the enrichment function with functions satisfying the partition of unity (PU) that results in a conforming approximation.

From a certain point of view, this enrichment technique can be seen as a new element type with special shape functions. The idea of constructing specific elements to model non smooth physical problem is not really new too. Among others, we can cite Barsoum [14] who proposed an element dedicated to obtain accurate singular solution around crack tip. While these techniques obtained effective results for some specific problems, the general scheme developed by Melenk and Babuska [110] is more attractive due to the fact that it consists in a generalization of the field approximation and that it can equivalently be applied to any numerical method including the FEM. Moreover, it is less demanding in terms of implementation in existing FE codes.

Since its introduction, the PU concept has focused a lot of interest and has been the topic of intensive researches and applications. In [163], Strouboulis and co-authors introduced the Generalized Finite Element Method (GFEM) for solving different elliptic problems by enriching the entire domain. The method was referred to GFEM since the classical FEM is a special case of this method. The enrichment technique improves the solution by introducing additional shape functions but the second advantage of this method is that discontinuous, singular or 'exotic' shape functions can be added allowing to represent non smooth behavior independently of the mesh. Later, as most non-smooth or non differentiable solution properties, such as jumps, kinks, and singularities are generally local phenomena, they introduced the idea of local enrichment by restricting the enrichment only to a subset of the domain.

Inspired by the possibilities of the GFEM, the first implementation of the so-called eXtended Finite Element Method (X-FEM) has been proposed by Belytschko and Black [19] where they enriched the finite element space with asymptotic crack-tip displacement field to treat 2D static fracture problems. In this seminal work, only the nodes close to the crack tips were enriched with the four functions spanning the near tip fields. At this stage, the method was able to model non conforming cracks tips and capturing accurate stress intensity factors with minimal mesh refinement. In Moës *et al.* [114] and Dolbow [52], the approach has been extended with the introduction of another enrichment function able to treat the discontinuity occurring across the crack lips. Hence, combining a Heaviside enrichment function and the crack tip field enabled modeling a complete crack embedded inside a non conforming mesh.

Following these pioneering works, the X-FEM approach has rapidly focused the interest of researchers and the method has been extended to arbitrary branched, intersecting cracks [51] and 3D cracks [166]. In [162], Stolarska *et al.* proposed to use the Level Set Method to represent the crack and to model the crack growth in 2D. Beside providing a theoretical method to update the position of the cracks, the use of the Level Set Method offered complementary capabilities such as simplifying the selection of the enriched nodes, defining the enrichment functions as well as localizing the interfaces. In [115], the method has been extended to 3D for non planar 3D crack growth. In [164], Sukumar *et al.* extended the X-FEM to model holes and inclusions. The X-FEM has obtained such promising results in fracture mechanics that some authors have immediately foreseen the opportunity of applying X-FEM to many kinds of problems in which discontinuities and moving boundaries arise. As example publications, we can mention Belytschko *et al.* [25] who applied the X-FEM to the modeling of composite fiber orientation in a micro-structure or Moës *et al.* [113] who presented a computational approach to study complex micro structures, Chessa *et al.* [44] who studied the case of two phase fluid problems. Applications to fluid structure interaction can also be found in Legay *et al.* [97] or Gerstenberger and Wall [74].

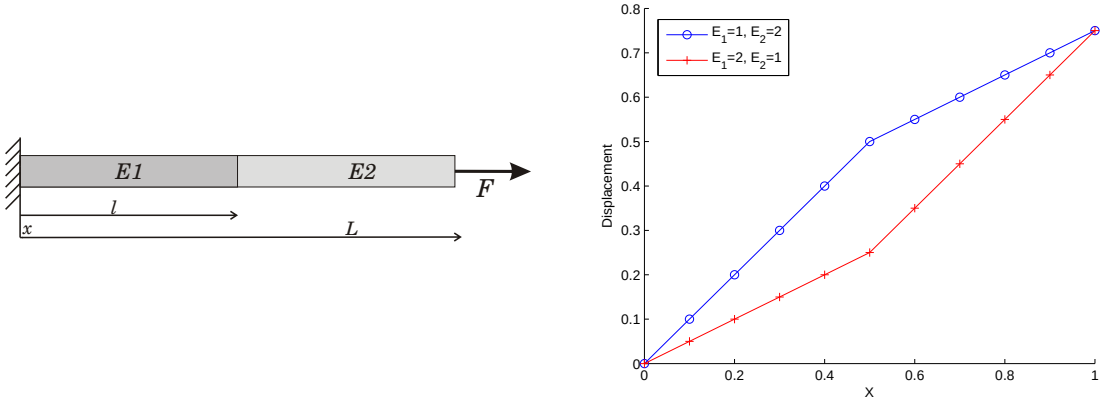
In the first implementation of the GFEM, the enrichment was realized on the entire domain and mainly applied for weak discontinuities or for enriching the approximation space with harmonic functions for instance. The X-FEM has been developed in parallel to the GFEM, with local enrichment devoted to crack simulation. Later, Strouboulis used the acronym GFEM in application using local enrichments with a discontinuity similar to X-FEM. Thus, X-FEM and GFEM are in fact almost identical and both derive from PUM. They mainly differs in the forms of the PU that is used and in the use of different local spaces. Several rather complete reviews on the X-FEM/GFEM have been published, some of them focus only on crack such as [21, 90, 140] while the others [1, 70] are more general and cover different application fields. A text book by Mohammadi has also been published [116].

## 3.2 Principles of the method

In the classical Finite Element Method, modeling discontinuities inside an element is not possible because the shape functions are generally at least $C^1$ on the element and $C^0$ between elements. Thus, if a kink or a jump is present inside a finite element, a $C^1$ shape function approximation is not able to reproduce the exact field and the kink or the jump is smeared out. Therefore, if discontinuities are present in the model, the discontinuity position has to coincide with the finite element mesh and a remeshing process is required each time the singularity position or its shape evolves in the structure/computational domain.

The X-FEM overcomes this restriction by adding to the classical FEM approximation some particular discontinuous, singular or ad-hoc shape functions through a local PU. Hence, these additional shape functions are directly related to the nature of the discontinuity. They allow the inclusion of cracks, material discontinuity, phase changes or any discontinuity that are not aligned with the mesh and avoid the expensive and delicate mesh generation stage. Moreover, the addition of proper shape functions to the space of the trial solutions allows a better convergence and overcomes the need for important mesh refinements near a singularity. In fact, the conformity dependence of the mesh, which is necessary for a FE model, is replaced by a geometrical dependence of the enrichment functions since the definition of these enrichments depend on the position of the discontinuity inside the computational domain.

To introduce the basic aspects of the enrichment techniques, we consider the example of the analysis of a bi-material 1D rod in traction. Let imagine that one wants to model a clamped-free bi-material rod of length L with two distinct Young moduli $E_1$=1 N/m$^2$ and $E_2$=2 N/m$^2$ subject to a traction force $F$=1 N along the $x$ axis (see Fig. 3.1 (a)).




(a) Geometrical model (b) Analytical solution

Figure 3.1: Bi-material rod in traction

The exact solution to this problem can be obtained analytically and is given by:

$$u(x) = \begin{cases} \frac{F}{E_1 A}x, & 0 \leq x \leq l \\ \frac{F}{E_2 A}x + (\frac{1}{E_1} - \frac{1}{E_2})\frac{Fl}{A}, & l < x \leq L \end{cases}$$

One can see in Figure 3.1 that the displacement field along the rod is not smooth and presents a kink (first order discontinuity) in $x = l$ where the material properties change. As a consequence the strain field $\epsilon(x)$ is discontinuous:

$$\epsilon(x) = \frac{\partial u}{\partial x} = \begin{cases} \frac{F}{E_1 A} & 0 \leq x \leq l \\ \frac{F}{E_2 A} & l < x \leq L \end{cases}$$

### 3.2.1 Finite Element Approximation

To analyze this rod with a FE method, we first have to construct the approximation space. In structural mechanics, each shape function $N_i$ is associated to a node $i$ and has a compact

support $\omega_i$ defined as the union of the element connected to the node $i$ (see Fig.3.2). In FEM, the approximation space is defined in terms of the nodal shape functions and described as:

$$u^h(x) = \sum_{i \in I} N_i(x) u_i$$

where $u^h$ corresponds to the approximate solution of the exact field $u$, $u_i$ is the nodal field value and $I$ is the set of all nodes in the computational domain.

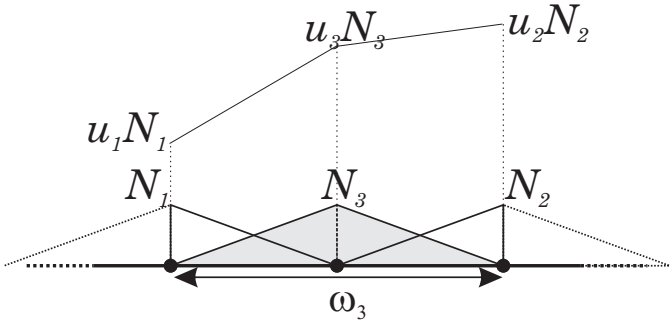



Figure 3.2: Finite element discretization of a one dimensional problem. Illustration of inter-element $C^0$ and inner-element $C^1$ continuity and $\omega_3$ the support of shape function of node 3.

With first order shape functions, the approximation is continuous and piecewise linear on the domain. Due to the nature of the shape functions, the approximation interpolates the values $u_i$, that is $u^h(x_i) = u_i$. The *interpolant property* is one key feature that facilitate the use of the FEM because the values $u_i$ have a physical meaning as they represent the displacement or the field value at the point $x_i$. Moreover, it is this characteristic that enables imposing strongly the essential Dirichlet boundary conditions. The linear precision comes from the fact that the shape functions are constructed such that:

$$\sum_{i \in I} N_i(x) = 1, \quad \sum_{i \in I} N_i(x) u_i = u$$

Hence, if we impose a linear nodal field, the approximation is able to model exactly this field. This shape function property is often referred to the reproducing property implying that the shape functions form a PU. This property corresponds also to the ability of the approximation to reproduce exactly a rigid body mode and constant strain states which is necessary to ensure the convergence and pass the patch test [187].

Using a linear approximation to model the bi-material rod, the approximation solution $u^h$ is piecewise linear. It is able to reproduce a kink only at inter-element boundaries where the approximation is $C^0$. From this fact, we know that a classical finite element approximation requires a minimum of two finite elements to model perfectly the problem under study. In fact, two FE is the minimum only if we ensure that their boundaries are conforming with the material discontinuity. Otherwise the approximation cannot reproduce the kink and the solution is smeared out. If only one element is used with linear shape functions, the accuracy is very poor (see curve with blue crosses in Figure 3.3). Using higher order approximations reduces the error as one can notice in Fig. 3.3 with circle and square markers line (order 2 and 3 shape functions respectively) but the discontinuity can not be retrieved exactly.

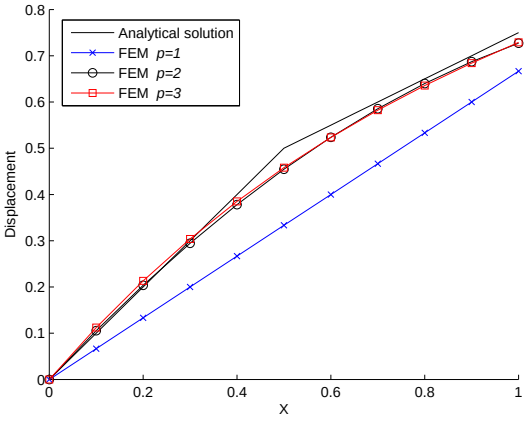



Figure 3.3: Finite element solution of the 1D bi-material rod with approximation of order $p$=1, 2 and 3.

### 3.2.2 Global enrichment and GFEM

From the above observations, we can state that due to the nature of the shape functions of a FEM approximation, the exact solution of a simple non smooth problem cannot be obtained unless we have previously build a tailored conforming mesh. Moreover, it clearly appears that, to get rid of the requirement to place a node on the discontinuity, the polynomial approximation space has to be expanded such that it can reproduce exactly the discontinuity inside an element. In order to highlight the advantage of the PUM method over different enrichment methods, we first introduce the idea of a global enrichment using the simplest approach consisting in adding an enrichment function to the FEM approximation.

First, the expanded approximation space has to be constructed and an enrichment function has to be chosen. It is obvious that this enrichment function should own the same characteristics than the field in order to be able to render the exact solution. In the case of the bi-material rod, the response is piecewise linear, possesses a kink where the change of material occurs and has a discontinuous field derivative. As a first approach to reproduce the kink in the displacement response, one can simply add a piecewise linear function with a kink. In this case, one defines an enrichment function $g(x)$ as:

$$g(x) \;=\; \begin{cases} \frac{x}{l}, & 0 > x \leq l \\ \frac{L-x}{L-l}, & l > x \leq L \end{cases}$$

And a first enriched approximation can be given by:

$$u^h(x) = \sum_{i \in I} N_i(x) u_i + a g(x)$$

The function $g(x)$ spas over the domain [0,L]. It is therefore called a global enrichment function and the degree of freedom '$a$' has a global influence on the response.

We first model the bi-material with 3 elements of length L/3 while it will be shown later that one element is enough. The discretized field over the domain is given by:

$$u^h(x) = u_1 N_1(x) + u_2 N_2(x) + u_3 N_3(x) + u_4 N_4(x) + a g(x)$$

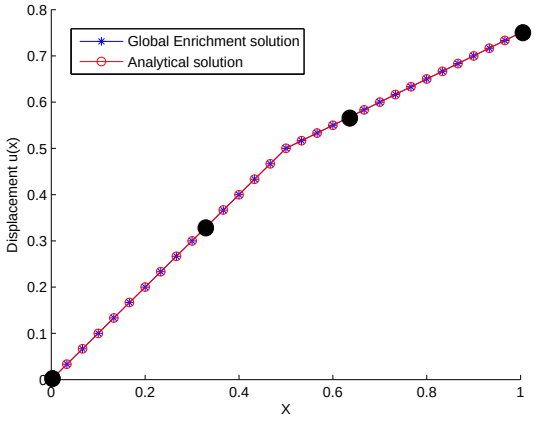



Figure 3.4: Global-Local solution of the bi-material rod

We can notice in Figure 3.4 that this approximation can retrieve the exact solution of the bi-material rod. However, this approximation, which can be referred as the Global-Local Method of Mote [119], is in fact not very well suited in practice. Indeed the enrichment function is global, which means that while the center element is the sole element that needs a specific enriched approximation space, the whole domain is enriched. Hence, it is not very efficient as we have to integrate unnecessary terms on these elements. Moreover, it also means that this approach destroys the banded structure of the stiffness matrix. This problem can simply be solved by setting the function value of $g(x)$ to be zero where the enrichment is not necessary. Hence, the enrichment becomes local and the stiffness matrix remains sparse. While it is not a problem in this 1D problem, a unique enrichment function and consequently a unique degree of freedom '$a$' generally introduces a lot of difficulties in higher dimensions because the construction of the enrichment function can become very arduous. Finally, this global enrichment does not possess the interpolant property of the FEM approximation as the displacement values at a node are not equal to $u_i$ unless in the trivial case where '$a$' equal zero or if the $g(x)$ function vanishes on nodes.

To circumvent these problems Melenk and Babuska [110] proposed the PUM. In this approach a set of degrees of freedom $a_i$ is introduced instead of a single parameter '$a$' and the enrichment function $g(x)$ is multiplied by the term $\sum_{i\in I^*} N_i^*(x)a_i$ leading to the following approximation:

$$u^h(x) = \underbrace{\sum_{i\in I} u_i N_i(x)}_{FEM} + \underbrace{\sum_{i\in I^*} a_i N_i^*(x) g(x)}_{Enrichment} \tag{3.1}$$

In [110] and [84] Melenk and Babuska developed the mathematical foundation of the PUM, demonstrating the efficiency of the method and including a proof of the convergence of PUM enriched approximation. A necessary condition to ensure the convergence requires that the functions $N_i(x)$ and $N_i^*(x)$ build a partition of unity over the analysis domain: $\sum_{i\in I} N_i(x) = 1$ and $\sum_{i\in I} N_i^*(x) = 1$. Thus, in the FEM, the functions $N_i^*(x)$ and $N_i(x)$ are chosen to be the classical the shape functions, and generally $N_i^*(x)$ and $N_i(x)$ are the same functions yielding to:

$$u^h(x) = \sum_{i\in I} N_i(x)(u_i + a_i g(x))$$

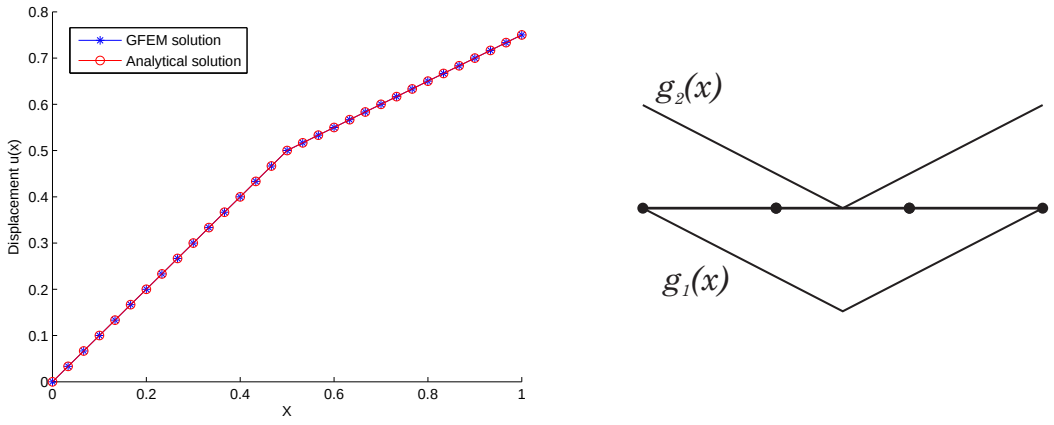
In this case, the sets of nodes $I$ and $I^*$ are identical and the number of degrees of freedom can grow considerably with respect to the size of the standard FEM approximation as the problem size doubles.

In the seminal papers on PUM [84, 110], there is no assumption on the approximation method and thus the $N_i(x)$ and $N_i^*(x)$ can be any functions that satisfies the PU. The multiplication by the $N_i^*(x)g(x)$ plays a crucial role for the convergence property of the method, but it also creates a local enrichment that allows to keep a sparse system of equations since the $N_i^*(x)$ shape functions are non-zero only on the compact support $w_i$ of node $i$. Notice that function $g(x)$ is multiplied by the standard shape functions $N_j(x)$ for three reasons:

- we get the important property of partition of unity for the function $g(x)$ because $\sum_{i \in I} N_i(x)=1$ and the enrichment function can be represented exactly.
- if only one enrichment function $g(x)$ is used for all the additional degrees of freedom (DOFs), the multiplication provides several $N_j(x)g(x)$ which are independent for each DOF;
- since the $N_i(x)$ shape functions are non-zero only on the compact support $w_i$ of node $i$, the system of equations remains sparse.

As one can notice in (3.1), in the most general case, the approximation does not have the Kronecker-$\delta$ property. Hence, $u^h(x_i) \neq u_i$ and the imposition of the essential boundary conditions is difficult as we have to evaluate all terms of the approximation to correctly enforce the boundary condition. This happens when the enrichment function $g(x)$ does not vanish on the element boundaries. Hence, by comparing the two different enrichments $g_1(x)$ and $g_2(x)$ depicted in Figure 3.5 (b), it is clear that it is preferable to use $g_1(x)$ as it vanishes on extremity nodes.

Using $g_1(x)$ as an enrichment function, the imposition of essential boundary conditions is straightforward and the solution obtained for the bi-material rod is depicted in Fig. 3.5 (a).




(a) GFEM solution of the bi-material rod (b) GFEM enrichment for the bi-material rod

Figure 3.5: GFEM solution and enrichment function of the bi-material rod

As previously mentioned, it can be difficult to build an enrichment function that vanishes on

the nodes where the boundary conditions are to be enforced. Thus, to recover the Kronecker-$\delta$ property on the whole domain, we have to construct carefully the enrichment function. A convenient procedure to obtain such enrichment is generally achieved by shifting the enrichment function by subtracting the nodal value $g(x_i)$ to the enrichment:

$$u^h(x) = \sum_{i \in I} u_i N_i(x) + \sum_{i \in I} N_i^*(x)(g(x) - g(x_i))a_i$$

This method was first proposed in [24], and it can be shown that this formulation can still reproduce the enrichment $g(x)$. However, one has to note that the enrichment is now equal to zero on the nodes but it is different from zero along the element boundaries which can be a problem to enforce essential boundary conditions along the FE boundaries.

### 3.2.3 X-FEM

As mentioned in the introduction, the PUM or GFEM has been initially introduced with a global enrichment for analyzing problems where non polynomial solutions take place on the entire domain. The X-FEM is derived directly form these methods but it has been introduced for the analysis of cracks or, more generally, for problems exhibiting local non-polynomial solutions. Hence, the X-FEM is generally regarded as a local PUM method and the main difference is that the enrichment is realized on a subset of elements to capture local phenomena. This means that the enrichment function is bounded to a local region of the computational domain where a specific behavior is likely to occur. Hence, it takes advantage from the enrichment technique of PUM/GFEM but it reduces the total number of degrees of freedom to limit the system dimension and to remain computationally efficient. The standard X-FEM approximation is stated in the following general form:

$$u^h(x) = \sum_{i \in I} N_i(x) u_i + \sum_{j \in J} N_j^*(x) a_j g(x)$$

where $I$ is the set of classical DOFs and $J$ a set of nodes that support the additional DOFs ($J \in I$). From this approximation, we can clearly see that only a subset of nodes are enriched compared to GFEM. Note that later, GFEM has also been used with a local enrichment and the distinction between these two methods has become quite tedious. In the global enrichment method, all elements are enriched and they all bears an equal number of extended degrees of freedom. Now, since only a subset of the nodes is enriched, one can find three types of elements in a meshed domain:

1. Classical FE with no additional degrees of freedom, that are used everywhere in the domain where the solution is expected to be smooth;

2. Fully enriched FE: these elements are generally named *reproducing elements* because on these elements, the $N_j^*(x)$ forms a partition of unity and they are able to reproduce exactly the enrichment function $g(x)$. It is in these elements that the discontinuity is located. In some applications, the zone of fully enriched elements can be enlarged to a geometrically defined neighboring zone around the discontinuity to obtain a better accuracy;

3. Partially enriched elements named *blending elements*: these elements support both classical and enriched degrees of freedom but not all the nodes support an enriched degree of freedom. In these elements, the $N_j^*(x)$ are non-zero and they do not form a PU. Therefore they cannot represent the function $g(x)$. They are used as *transition* elements between reproducing elements and the normal elements.

At first glance, the idea of local enrichment seems computationally more efficient than global enrichment on the whole domain. However, this is not generally true because the presence of the blending elements, which depend on the enrichment function characteristics, can considerably deteriorate the accuracy of the solution as we shall see in the following.

### Blending

As mentioned earlier, the choice of the additional shape functions is directly related to the nature of the field behavior. However, the choice of the enrichment is very important in order to avoid strange behaviors or sub-optimal rates of convergence. To illustrate a good choice of an enrichment function, we present and discuss the X-FEM analysis of the bi-material rod with a first "naive" local enrichment. Then, this enrichment is modified to remove the identified problems and finally an adapted enrichment is presented.

#### *First enrichment: abs-enrichment*

The first enrichment function to model a discontinuous gradient could be defined as:

$$g(x) = |L(x)| \equiv |\phi(x)| \tag{3.2}$$

where $L(x)$ is the distance of a point $x$ to the location of the material change or, equivalently, the absolute value of the Level Set $\phi(x)$ at point $x$ if the discontinuity is modeled with a signed distance function. This $g(x)$ function forms a reversed *hat* over the element.

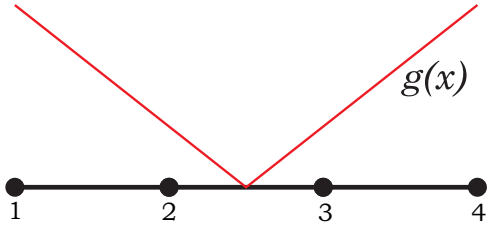



Figure 3.6: *abs*-enrichment function

Similarly to the previous example with GFEM, using this type of enrichment with only one element makes difficult to impose the essential boundary conditions. The application of the loads is also affected as the enrichment does not vanish on the nodes of the reproducing element. To illustrate also the effect of blending elements, we use three elements of length $\frac{L}{3}$ to model the rod with a material discontinuity located at $l = \frac{L}{2}$. However, as the enrichment function is non zero on the boundaries of the center element, the total displacement on the center element is made of the contribution of degrees of freedom $u_i$ and $a_j$. So, to fulfill the continuity condition at inter-element boundaries, we cannot set the enrichment to zero on the other neighboring elements and the $g$ function has to be extended to the first and third elements. Otherwise one would have a discontinuous displacement at element interfaces (1-2 and 2-3):

$$\underbrace{u(\frac{L}{3}) = u_1}_{\text{On the first element}} \quad \textit{and} \quad \underbrace{u(\frac{L}{3}) = u_1 + a_1}_{\text{On the second element}}$$

However, to limit the number of degrees of freedom, the node 1 and 4 are not enriched and there are two types of elements: 2 blending elements and 1 reproducing element. This model leads to

the following X-FEM discretization:

$$u^h(x) = \begin{cases} u_1(1-\frac{x}{L}) + u_2\frac{x}{L} + a_1\frac{x}{L}\frac{L+l-x}{l}, & 0 \leq x \leq \frac{L}{3} \\ u_2(1-\frac{x}{L}) + u_3\frac{x}{L} + a_1(1-\frac{x}{L})\frac{l-x}{l} + a_2\frac{x}{L}\frac{l-x}{l}, & \frac{L}{3} \leq x \leq l \\ u_2(1-\frac{x}{L}) + u_3\frac{x}{L} + a_1(1-\frac{x}{L})\frac{x-l}{L-l} + a_2\frac{x}{L}\frac{x-l}{L-l}, & l \leq x \leq \frac{2L}{3} \\ u_3(1-\frac{x}{L}) + u_4\frac{x}{L} + a_2(1-\frac{x}{L})\frac{L-l+x}{L-l}, & \frac{2L}{3} \leq x \leq L \end{cases}$$

Hence, although the $g(x)$ function is not zero on the first node, there is no difficulty in enforcing the essential boundary conditions $u^h(x{=}0){=}0$ as the first element is a blending element and the node 1 is not enriched. The application of this enrichment to the bi-material rod gives the results presented in Fig 3.7.

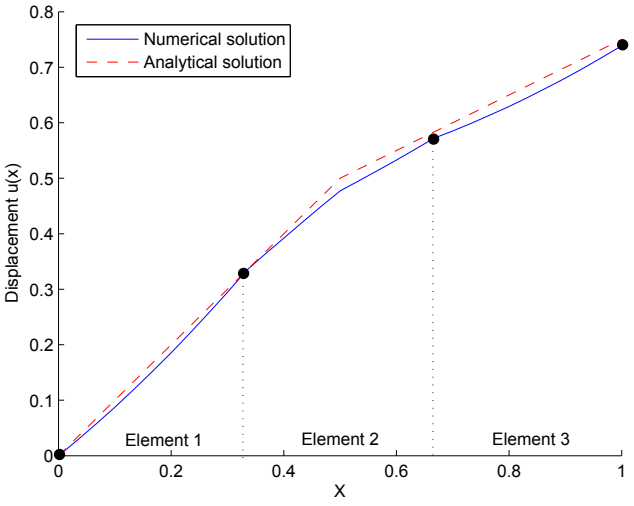



Figure 3.7: Solution to the bi-material rod using the *abs*-enrichment

We can see in Figure 3.7 that this approximation fails at reproducing the analytical solution. This error is due to the presence of the blending elements. We notice that on the reproducing element (element 2), because the property of the partition of unity is respected, the enrichment can be reproduced exactly and the solution is piecewise linear as expected. However, in the blending elements (elements 1 and 3), the introduction of a partial enrichment deteriorates the solution. In fact, this is due to a non complete basis function set. On these elements, the approximation is made of two linear functions and one second order shape function. Hence, it can be noticed that, in the non trivial case $a_j{=}0$, the $N_j(x)g(x)$ function introduces a parasitic term (a quadratic contribution) on the first and last elements that prevents the solution to match the analytical solution. As the enrichment is partial, there are no way to compensate the quadratic contribution as it was the case with GFEM. Thus, even if this enrichment presents a discontinuous derivative, it is not well adapted to model fields with discontinuous gradient.

Compared to the GFEM model, two quadratic functions are present on the first element and the modulation of the enriched degrees of freedom cancels this effect. Thus, the problem in the blending elements does not occur in the GFEM because the enrichment is realized globally and all elements can be considered as reproducing. There is not lack of partition of unity. Here, on the blending elements, the $N_i^*$ forms a partition of unity only on the reproducing element. However, the problem in blending elements would not occur if the enrichment is piecewise constant

because the order of the classical FEM approximation should not be modified and no parasitic term would be added to the approximation. Hence, when the enrichment is constant such as in an Heaviside enrichment, the order of the enriched approximation part is not higher than the standard FE part.

This problem has been reported in early X-FEM publications, by Sukumar *et al.* [166] who obtained suboptimal convergence rates with the *abs*-enrichment for discontinuous derivatives in two bi-material plates. Thus, different techniques have been developed to suppress this behavior. Chessa *et al.* [46] propose an Enhanced Strain approach based on the Hu-Washizu principle that eliminates the unwanted terms in the blending elements by designing an adequate enhanced strain field. In the same paper [46], they also propose to use hierarchical elements for polynomial enrichments. In [71], Fries propose another method which is more general and easy to implement. The idea is to completely enrich the blending elements and multiply the enrichment functions with a ramp function decreasing to zero over these elements. This method can be applied to any enrichment type and is very efficient. Finally, another approach to remove the problem in the blending element, called *Local PUM* and also introduced by Fries, is to construct the enrichment function such that it vanishes on the nodes of the reproducing element as presented hereafter.

***Second enrichment: Local PUM***

Based on the previous observations, it is obvious that a more appropriate solution consists in building an enrichment function which vanishes at the nodes of the elements, similar to a classical bubble shape functions (i.e. as in 9 nodes quadrilateral element [49]).

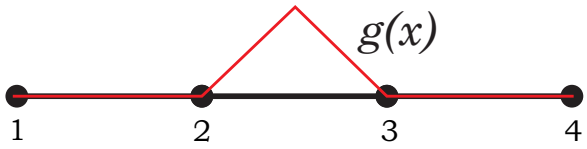



Figure 3.8: X-FEM bubble enrichment function

That is, one can take the previous $g(x)$ function and reverse it to get a *hat* function which does not require blending elements anymore. Therefore, there is no problem to treat the boundary conditions and we can discretize the displacement field as following:

$$u^h(x) = u_1 N_1(x) + u_2 N_2(x) + u_3 N_3(x) + u_4 N_4(x) + a_2 N_2(x) g(x) + a_3 N_3(x) g(x)$$

where

$$g(x) \quad = \quad \begin{cases} 0, & 0 \leq x \leq \frac{L}{3} \\ \frac{x}{l}, & \frac{L}{3} \leq x \leq l \\ \frac{L-x}{L-l}, & l \leq x \leq L \\ 0, & \frac{2L}{3} \leq x \leq L \end{cases} \tag{3.3}$$

This discretization is able to model exactly the analytical solution for all positions of the material interface, which is not surprising because this simple example is similar to the GFEM with a localized enrichment. Moreover, as the value of this shape function (3.3) vanishes at all nodes, there is no difficulty with the application of the essential boundary conditions. Therefore, the bi-material rod can be modeled using just one single element using this simplified approximation:

$$u^h(x) = u_1 N_1(x) + u_2 N_2(x) + a_1 N_1(x) g(x) + a_2 N_2(x) g(x)$$

where

$$g(x) = \begin{cases} \frac{x}{l}, & 0 \leq x \leq l \\ \frac{1}{L-l}(L-x), & l \leq x \leq L \end{cases}$$

and the solution is represented in Figure 3.9.

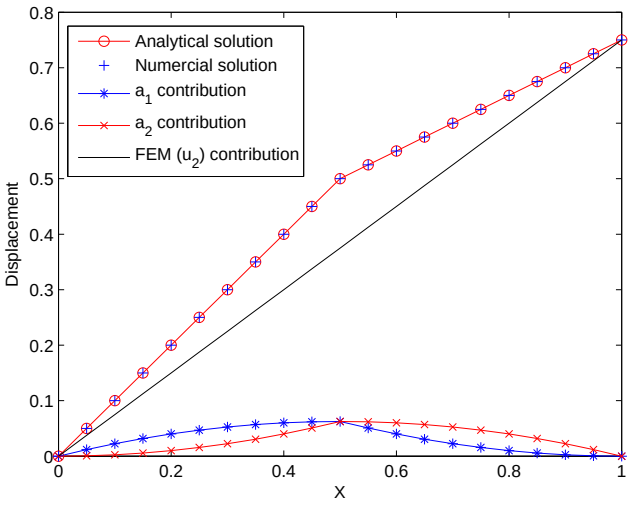



Figure 3.9: Solution with X-FEM and bubble enrichment function and 1 element

The added shape functions are quadratic but it is interesting to see that the enriched component of the displacement gets the following expression:

$$a_1 N_1 g(x) + a_2 N_2 g(x) = \begin{cases} a_1(\frac{x}{l} - \frac{x^2}{lL}) + a_2 \frac{x^2}{lL}, & 0 \leq x \leq l \\ \frac{1}{L(L-l)}(a_1(1 + \frac{x^2}{L^2} - 2\frac{x}{L}) + a_2(\frac{x}{L} - \frac{x^2}{L^2})), & l \leq x \leq L \end{cases}$$

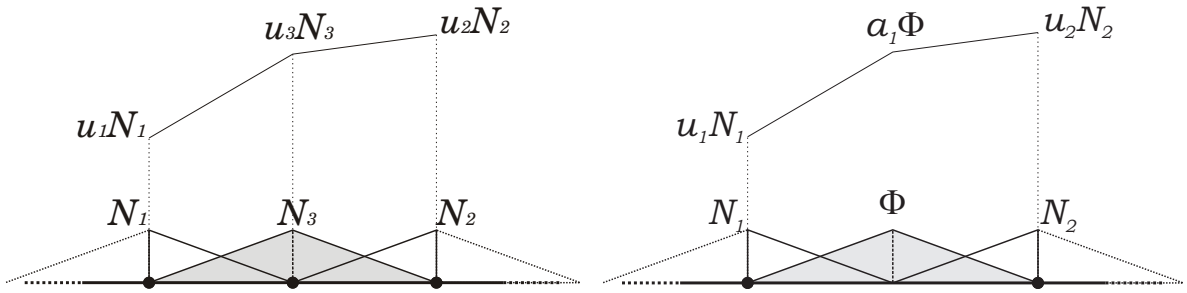
which leads, if $a_1$=$a_2$=$a$, to a piecewise linear function corresponding to the $g(x)$ function:

$$a_1 N_1 g(x) + a_2 N_2 g(x) = a(N_1 + N_2)g(x) = ag(x) = \begin{cases} a_1 \frac{x}{l}, & 0 \leq x \leq l \\ a_1 \frac{1}{L-l}(L-x), & l \leq x \leq L \end{cases}$$

This demonstrates that the enrichment is able to represent a linear displacement over the element. Additionally, we can also add that this element recovers a uniform deformation when $E_1$=$E_2$ (leading to $a_i$=0), allowing to pass the constant strain patch test.

### Remark

As we have noticed above, the discontinuity is accurately modeled when the two enriched degrees of freedom $a_i$ are equals. This results from the fact that only one enriched extra DOF is sufficient to model the discontinuous gradient. We can understand this idea if we look at the shape functions used for the two finite elements model in Fig. 3.10 (a). We can observe that the solution is given by a linear contribution of node 2 and node 1 (in the case of $u_1 \neq 0$) and that the kink is ruled by the contribution of node 3.




(a) FEM Shape functions (b) X-FEM Shape Functions

Figure 3.10: FEM and X-FEM shape functions

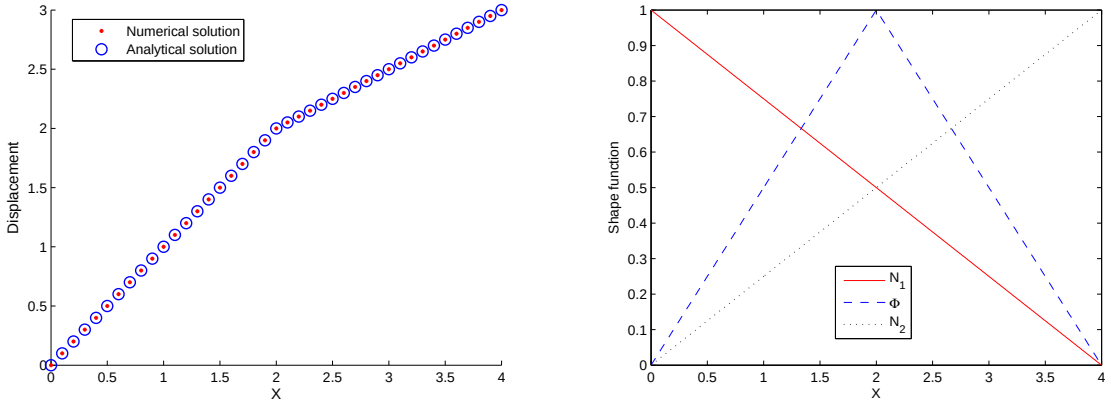
From these observations, we can define an appropriate function to add to a X-FEM element in order to get a space able to represent the solution with 2 classical nodes and an additional one '$a_1$'. Therefore if we discretize the displacement field with the following equation:

$$u^h(x) = u_1 N_1(x) + u_2 N_2(x) + a_1 g(x)$$

where

$$g(x) \quad = \quad \begin{cases} \frac{x}{l}, & 0 \leq x \leq l \\ \frac{L-x}{L-l}, & l \leq x \leq L \end{cases} \tag{3.4}$$

the singularity is introduced by the *hat* function $g(x)$ which is similar to the *abs*-enrichment of (3.2). In Figure 3.11, we plot the results obtained with these tests functions and one gets obviously the right solution.




(a) Comparison between analytical and X-FEM 3-nodes element solution (b) Shape functions defined on the beam

Figure 3.11: Solution of the 3-nodes X-FEM element

These results are very interesting as the number of enriched nodes and then the total number of degrees of freedom are reduced. Likewise, only first degree shape functions are used in (3.4) and the property of partition of unity still holds. However, in practice, this enrichment can not be used because its generalization to 2D and 3D is not always possible. In fact, if we consider

a triangular element which presents a material interface parallel to one of its edge, we can build a bi-planar shape function that connect an edge on one side and a node on the other side (see Fig. 3.12). However, if the interface is not either parallel to the side we cannot find a plane connecting 4 points that are not coplanar and the generalization to 2D is therefore not possible.

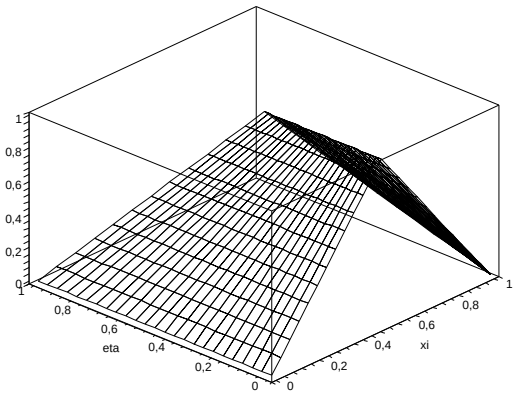



Figure 3.12: Possible enrichment for a triangle X-FEM element

## 3.3 Usual enrichment types

The discontinuities in a model can take different forms and they can be classified as weak or strong discontinuities. The weak discontinuities occur when the gradient of the field solution exhibits a discontinuous behavior. There is a kink in the solution. This type of weak discontinuity appears for instance across different material interfaces. The second type of discontinuities, i.e. strong discontinuity, occurs when the solution field is discontinuous and it appears for instance in the vicinity of cracks and of material-void interfaces. To model these types of discontinuities, specific enrichments have to be constructed and they are presented in the following.

### 3.3.1 Modeling fields with discontinuous gradient

The first enrichment to model discontinuous gradient has been proposed by Sukumar *et al.* [164] for modeling 2D internal inclusions with X-FEM. In this paper, the authors proposed an enrichment function $g(x)$ given by:

$$g(x) = |\phi(x)|$$

which is similar to the *abs*-enrichment of the previous bi-material rod example. They also pointed out the error introduced by this kind of enrichment (due to blending elements) and proposed to extend it differently in the layer of elements surrounding the element in which the discontinuity is located. That is, they proposed the two following solutions:

1. Extend the $g(x)$ function such that the gradient of $g$ is minimized over the layer elements. Hence, the effects of the blending elements is minimized;

2. Extend the $g(x)$ function with a constant value. This solution is obviously the best as the layer elements now support a linear enrichment function which allows to capture the analytical solution.

Later, Chessa *et al.* [44] presented another enrichment function $N_j^e(\mathrm{x})$ depending on the node enrichment initially proposed for multi-phase problems and defined as:

$$u^h(x) = \sum_{i \in I} N_i(x) u_i + \sum_{j \in J} N_j^e(x) a_j$$

with

$$N_j^e(x) = N_j^*(x) g_j(x), \quad g_j(x) = |\phi_j| - |\phi(x)|$$

where $\phi_j$ is the Level Set value of node $j$ and $\phi(x)$ is the Level Set Value at point $x$ (see Fig. 3.13) and $N_j^*(x)$ is a classical FEM shape function that can be equal to $N_i(x)$.

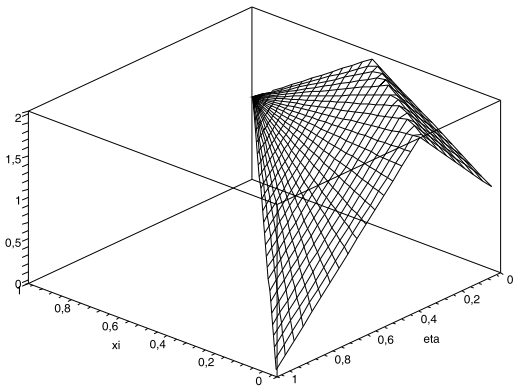



Figure 3.13: $g$ function of the node 3

However, this enrichment does not remove the problem encountered with the previous enrichment since blending elements are also present. Finally, Moës *et al.* [113] introduced a third enrichment (called *ridge* function), which is unique for all nodes of the element and is defined as:

$$u^h(x) = \sum_{i \in I} N_i(x) u_i + \sum_{j \in J} N^*(x) g(x) a_j$$

where

$$g(x) = \sum_i |\phi_i| \, N_i(x) - \left| \sum_i \phi_i N_i(x) \right| \tag{3.5}$$

is represented in Figure 3.14.

One can first notice that this enrichment function (3.5) corresponds to the generalization of the last enrichment used in the 1-D rod problem. Then, among all the classical enrichment functions, this one remains the simplest as only the elements containing the interface needs an enrichment. Moreover, initially, this additional function was the only one able to reach the analytical solution in the 1-D trivial case and to recover the optimal rate of convergence $O(h)$ at least in 2D whereas all the other enrichments give rise to a poor rate of convergence. Later, the corrected X-FEM [71] and the intrinsic X-FEM [69] methods proposed by Fries and co-workers have proven to be efficient to obtain optimal rates of convergence for weak discontinuities too.

### 3.3.2 Modeling discontinuous fields

Discontinuous displacement fields occur in presence of cracks for instance. In that case the field present a jump in the response where the discontinuity lies. The field variables are decoupled

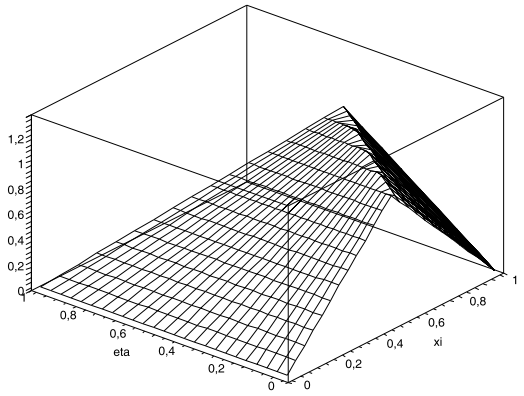



Figure 3.14: Unique $g(x)$ function

on both sides of the interface and their gradients can also be discontinuous across the interface. Such a field is obtained with a FEM by meshing explicitly the discontinuity so that edges or faces coincide with the crack and nodes must be placed on each side of the crack to allow a material separation along the crack surface.

### 3.3.3 Heaviside Enrichment

The first discontinuous enrichment that has been proposed by Moës *et al.* in [114] consists in adding the Heaviside step function $H(x)$ to model the discontinuity that arises along the crack faces. The standard FEM approximation is then extended with discontinuous shape functions as following:

$$u(x) = \sum_{i \in I} u_i N_i(x) + \sum_{j \in J} a_j N_j^*(x) H(x)$$

The $N_j^*(x)H(x)$ are the discontinuous shape functions constructed by multiplying a classical shape function $N_j^*(x)$ with a Heaviside function $H(x)$, which is equal to +1 on one side of the crack and -1 on the other side (see Fig. 3.15).

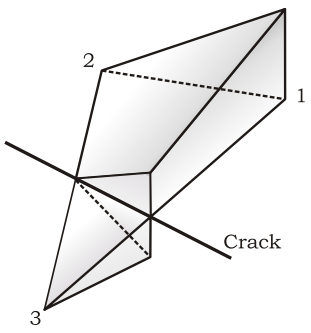



Figure 3.15: Representation of the extended shape function $N_1^*(x)H(x)$ of node 1 for a cracked element

Two definitions can be used for the Heaviside function:

$$H_1(x) = \begin{cases} 0 & x \in \text{Positive Level Set domain} \\ 1 & x \in \text{Negative Level Set domain} \end{cases} \tag{3.6}$$

$$H_2(x) = \begin{cases} -1 & \text{if} \quad x \in \text{Positive Level Set domain} \\ 1 & \text{if} \quad x \in \text{Negative Level Set domain} \end{cases} \tag{3.7}$$

The difference between the two Heaviside functions is that in the first case of (3.6), on the side where $H_1(x)$ is equal to zero, the enrichment contribution is equal to zero and the classical part of the approximation is equal to the total field. On the other side of the discontinuity, where the Heaviside function is equal to 1, the total field is given by $u + a$, and $a$ thus corresponds to the jump. In (3.7) the classical FEM approximation corresponds to the mean value and the enrichment is the jump that change its sign across the discontinuity.

### 3.3.4 Cracks

In presence of a crack inside the structure, two different non smooth behaviors occur in the structure. The first one is the presence of a strong displacement discontinuity across the crack face. This is modeled with an step enrichment as mentioned above. The latter is a field singularity that occurs at the tip. The introduction of enrichment functions to model cracks inside an element started with the pioneer work of Belytschko and Black [19] where they used the near-tip asymptotic field within a partition-of-unity method to model the crack tip singular field based on the analytical solution as:

$$u(x) = \sum_{i \in I} u_i N_i(x) + \sum_{k \in K} N_k(x) \left( \sum_l b_k^l F^l(x) \right)$$

So a fully embedded crack approximation is given by:

$$u(x) = \sum_{i \in I} u_i N_i(x) + \sum_{j \in J} a_j N_j(x) H(x) + \sum_{k \in K} N_k(x) \left( \sum_l b_k^l F^l(x) \right)$$

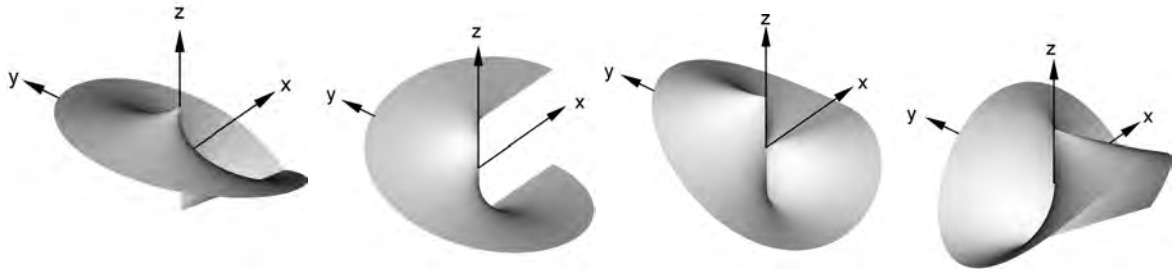
where

$$F^l(r, \theta) = [\sqrt{r} \sin(\frac{\theta}{2}), \sqrt{(}r) \cos(\frac{\theta}{2}), \sqrt{(}r) \sin(\frac{\theta}{2}) \sin(\theta), \sqrt{(}r) \cos(\frac{\theta}{2}) \sin(\theta)]$$

is the set of functions describing the singular displacement at crack tip and $K$ is the set of nodes to be enriched with the $F^l$ function (illustrated in Figure 3.16). The set $J$ is composed of the nodes whose support are entirely cut by the discontinuity whereas $K$ is made of the nodes such that their supports contain the crack tip. A representation of the different enriched nodes and the relative enrichment is depicted in the Figure 3.17. The squares denotes the nodes enriched with the Heaviside shape functions and the circles are the nodes enriched with the crack tip function.

The Level Set representation is very useful when dealing with crack tip shape functions. As these ones are defined through a local co-ordinate system ($r$,$\theta$) defined at crack tip, each crack growth needs a new definition of this referential. Therefore, the usual representation of cracks needs two Level Sets placed such that one Level Set $\psi$ model the crack itself and a second $\phi$ (orthogonal to the previous one) to represent the crack tip. This geometrical representation is very convenient as it builds a natural system of coordinates (see Fig. 3.18) and the value of $r$ and $\theta$ can be immediately obtained from the nodal value of these Level Sets.

$$\theta(x) = \arctan \frac{\psi(x,t)}{\phi(x,t)}, \quad r = \sqrt{\psi^2(x,t) + \phi^2(x,t)}$$




(a) $F^1$ function (b) $F^2$ function (c) $F^3$ function (d) $F^4$ function

Figure 3.16: Representation of the extended shape function at crack tip

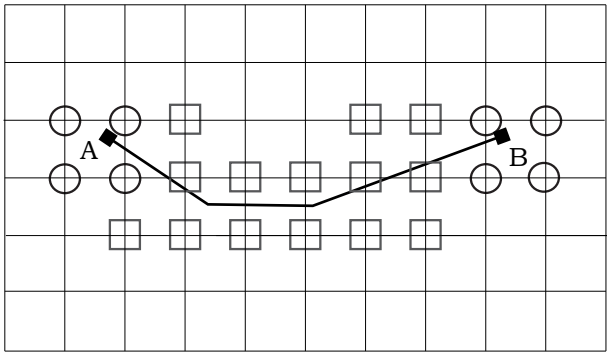



Figure 3.17: Representation of the enriched nodes for an embedded crack

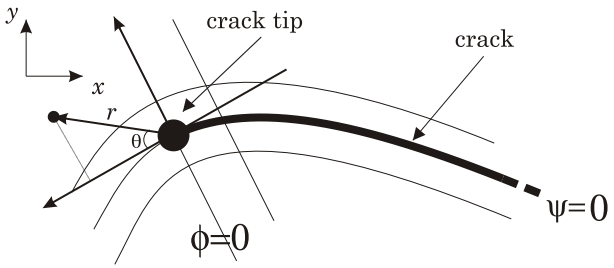



Figure 3.18: Representation of crack and crack tip

The modification of the displacement field approximation does not introduce a new form of the discretized finite element equilibrium equation but rather leads to an enlarged problem (see [114] and [166] for details on the governing equations). By taking only the Heaviside enrichment, one get:

$$\mathbf{Ku} = \mathbf{f} \Leftrightarrow \begin{bmatrix} K_{uu} & K_{ua} \\ K_{au} & K_{aa} \end{bmatrix} \begin{bmatrix} u \\ a \end{bmatrix} = \begin{bmatrix} f_u^{ext} \\ f_a^{ext} \end{bmatrix}$$

As the elements can now present discontinuous shape functions, the numerical integration scheme has to be modified in order to take care of the discontinuity. To use the classical Gauss numerical integration schemes, the elements embedding a singularity are divided into sub-triangular meshes coincident with the discontinuity boundary. Over this working mesh, a quadrature integration rule can be applied (see section 4.1 for more details).

### 3.3.5 Holes

Modeling material-void interfaces with X-FEM (see [164] and [51]) differs only marginally from the cracked structure case (i.e. the discontinuous field is also the displacement). Hence, it is possible to model material-void interface with a Heaviside function and appropriate Lagrange multipliers at the interface. However, in our implementation, we implemented the classical form of X-FEM for void inclusions where the displacement field is approximated by:

$$u(x) = \sum_i u_i N_i(x) V(x)$$

where

$$V(x) = \begin{cases} 1 & \text{if} \quad x \in \text{material zone} \\ 0 & \text{if} \quad x \in \text{void} \end{cases}$$

One can notice that this function $V(x)$ models perfectly the singularity in the displacement field inasmuch the function $V(x)$ cancels the displacement field in the void (see Fig. 3.19).

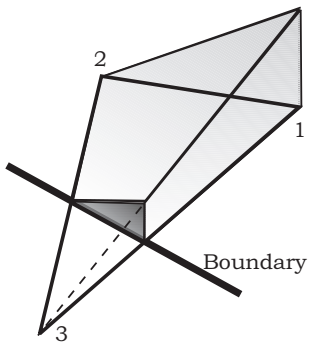



Figure 3.19: Representation of the shape function of node 1 on a cut element

In practice, the function $V(x)$ is not really introduced and the integration is only processed in the solid part of the element while the empty elements are removed from the formulation when building the stiffness matrix as we will see in the next chapter. Also, contrary to the two previous cases, the governing equation does not enlarge the system of equation of the finite element model.

Modeling holes with the X-FEM is a very appealing method for the shape optimization but also for the topology optimization as no remeshing is needed. Additionally, no approximation is

made on the nature of the voids conversely to numerous methods, e.g. the power penalization of intermediate densities (SIMP) method used in topology optimization which replaces void by a very soft material (see [27]).

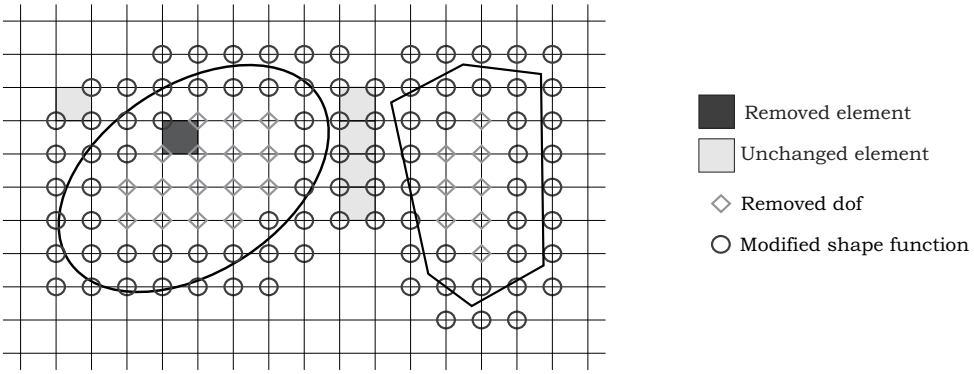



Figure 3.20: Representation of removed DOFs and removed elements when assembling structural matrices

## 3.4 Intrinsic X-FEM

In [69], Fries and Belytschko have introduced the notion of intrinsic enrichment and extrinsic enrichment methods. Hence, in this chapter, all the previously presented enriched method fall into the class of extrinsic enrichment as they rely on the PUM concept and they add special enrichment functions to a standard approximation space. In [69], the authors developed a method that does not add any additional unknowns and enrichment functions but rely on a modified approximation space. The computational domain is divided into overlapping domains and a PU approximation is built for each one. Thus, in the vicinity of a singularity, each node supports a specific shape function while the remaining nodes support standard FE shape functions. The different domains are coupled by using a ramp function to combine the different PU and to obtain only a single shape function per node. This method exhibits interesting results for strong and weak discontinuities and does not suffers from the blending problem.

## 3.5 X-FEM for other problems

We have limited the presentation of X-FEM to mechanical problems in this introduction. However, the enrichment strategy can be equally applied to other physical problems. For instance we can cite Chessa who adopted a enrichment function similar to the bi-material example for solidification problems in [45] and two phase fluids problems in [44].

The simple examples presented have illustrated the idea of introducing specific shape functions to model particular behaviors. Moreover, we can already remark that the integration of such a simple model has to be carried out carefully since the enriched shape functions present discontinuities. The integration has to be modified on the blending elements as they support extended shape functions leading to higher order shape functions.

Chapter 4

# Implementation of X-FEM for material-void interfaces and bi-material interfaces

The chapter 3 presented the general aspects of the X-FEM for various kinds of discontinuities. We now focus on the practical aspects of the method such as numerical integration, numerical accuracy, convergence, computation of mechanical responses and visualization. This chapter is mainly dedicated to the material-void interface element while bi-material interface element is mentioned at the end.

## 4.1 Numerical integration

### 4.1.1 Integration of a 2D isoparametric linear triangle finite element

In this section, we briefly remind the integration technique used in a classical mechanical finite element (FE). For the sake of simplicity, a first degree triangle element is taken as example, but the method can be transposed to other types of elementary topologies or degrees of approximation.

The displacement field $u$ is approximated with the following expression:

$$u(\mathbf{x}) = \sum_{i}^{n} N_i(\mathbf{x}) u_i$$

where $u_i$ is the nodal displacement of node $i$, $N_i$ the corresponding shape function and $n$ the total number of shape functions $N_i$. When defined in the reference space of local variables ($\xi$,$\eta$) (see Fig. 4.1 and Ref. [186]), the shape functions are given by:

$$N_1 = 1 - \xi - \eta, \qquad N_2 = \xi, \qquad N_3 = \eta$$

The displacement field is then expressed in terms of the nodal degrees of freedom ($u_i$,$v_i$) and the relative shape functions as following:

$$u = N_i u_i = \begin{bmatrix} N_1 & 0 & N_2 & 0 & N_3 & 0 \\ 0 & N_1 & 0 & N_2 & 0 & N_3 \end{bmatrix} \begin{Bmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ u_3 \\ v_3 \end{Bmatrix}$$

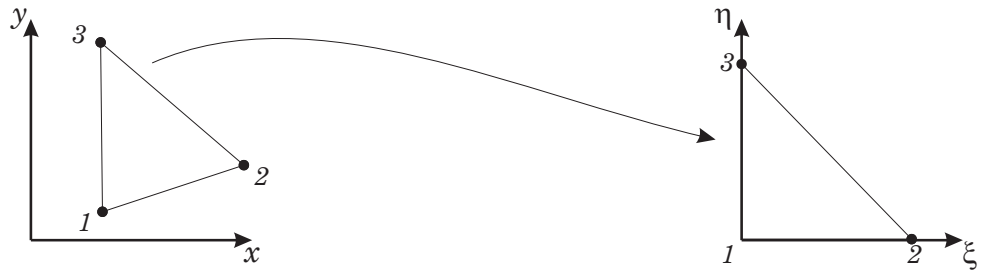



Figure 4.1: Triangle physical space and reference space

Thus, the symmetric linear deformation tensor is given by:

$$\begin{aligned}
\varepsilon &= Bu = \begin{bmatrix} \frac{\partial u}{\partial x} & 0 \\ 0 & \frac{\partial v}{\partial y} \\ \frac{\partial u}{\partial y} & \frac{\partial v}{\partial x} \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{bmatrix} \begin{bmatrix} N_1 & 0 & N_2 & 0 & N_3 & 0 \\ 0 & N_1 & 0 & N_2 & 0 & N_3 \end{bmatrix} \begin{Bmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ u_3 \\ v_3 \end{Bmatrix} \\
&= \begin{bmatrix} \frac{\partial N_1}{\partial x} & 0 & \frac{\partial N_2}{\partial x} & 0 & \frac{\partial N_3}{\partial x} & 0 \\ 0 & \frac{\partial N_1}{\partial y} & 0 & \frac{\partial N_2}{\partial y} & 0 & \frac{\partial N_3}{\partial y} \\ \frac{\partial N_1}{\partial y} & \frac{\partial N_1}{\partial x} & \frac{\partial N_2}{\partial y} & \frac{\partial N_2}{\partial x} & \frac{\partial N_3}{\partial y} & \frac{\partial N_3}{\partial x} \end{bmatrix} \begin{Bmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ u_3 \\ v_3 \end{Bmatrix} \\
&= \begin{bmatrix} B_{11} & 0 & B_{12} & 0 & B_{13} & 0 \\ 0 & B_{21} & 0 & B_{22} & 0 & B_{23} \\ B_{21} & B_{11} & B_{22} & B_{12} & B_{23} & B_{13} \end{bmatrix} \begin{Bmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ u_3 \\ v_3 \end{Bmatrix}
\end{aligned} \tag{4.1}$$

And finally, the stiffness matrix is defined by:

$$K = \int_V B^T H B \, dV = \int \int B^T H B \, |J| \, t \; d\xi d\eta \tag{4.2}$$

where $B$ is the matrix of the shape function derivatives and $t$ is the thickness. Because the shape functions $N_i$ are expressed in terms of $\xi$ and $\eta$, not $x$ and $y$, derivatives in (4.1) are not immediately available, the relation between the gradients between the two systems of coordinates must be established. Using chain rule, we can express the following relation:

$$\begin{cases} \frac{\partial}{\partial x} = \frac{\partial}{\partial \xi}\frac{\partial \xi}{\partial x} + \frac{\partial}{\partial \eta}\frac{\partial \eta}{\partial x} \\ \frac{\partial}{\partial y} = \frac{\partial}{\partial \xi}\frac{\partial \xi}{\partial y} + \frac{\partial}{\partial \eta}\frac{\partial \eta}{\partial y} \end{cases}$$

which, in matrix notation, gives:

$$\begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix} = \begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \eta}{\partial x} \\ \frac{\partial \xi}{\partial y} & \frac{\partial \eta}{\partial y} \end{bmatrix} \begin{bmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \end{bmatrix}$$

Unfortunately, the partial derivative of $\xi$ and $\eta$ with respect to $x$ and $y$ are not directly available, which is not the case of the inverse relation:

$$\begin{bmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix} \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix}$$

where the partial derivatives of $x$, $y$ with respect to $\xi$ and $\eta$ are available. Defining $[J]$, the Jacobian of the transformation between the spaces ($\xi$,$\eta$) and ($x$,$y$):

$$[J] = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix}$$

and its determinant $|J|$, it comes:

$$|J| = \frac{\partial x}{\partial \xi}\frac{\partial y}{\partial \eta} - \frac{\partial y}{\partial \xi}\frac{\partial x}{\partial \eta} \tag{4.3}$$

We obtain the relation between the gradients in $(x, y)$ and $(\xi, \eta)$:

$$\begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix} = [J]^{-1} \begin{bmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \end{bmatrix} = \frac{1}{|J|} \begin{bmatrix} \frac{\partial y}{\partial \eta} & -\frac{\partial y}{\partial \xi} \\ -\frac{\partial x}{\partial \eta} & \frac{\partial x}{\partial \xi} \end{bmatrix} \begin{bmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \end{bmatrix} \tag{4.4}$$

The equations (4.4) and (4.3) are then used to evaluate the components of the $B$ matrix and the determinant of the Jacobian. In an isoparametric FE, the geometry of an $n$ nodes element is defined through its nodal coordinates and the associate standard FEM shape functions:

$$x = \sum_i^n N_i(\xi, \eta) x_i, \quad y = \sum_i^n N_i(\xi, \eta) y_i$$

The Jacobian is directly obtained from nodal coordinates and shape function derivatives in reference space $(\xi, \eta)$:

$$[J] = \begin{bmatrix} \sum_i^n \frac{\partial N_i}{\partial \xi} x_i & \sum_i^n \frac{\partial N_i}{\partial \xi} y_i \\ \sum_i^n \frac{\partial N_i}{\partial \eta} x_i & \sum_i^n \frac{\partial N_i}{\partial \eta} y_i \end{bmatrix}$$

For $B_{11}$, if one uses the definition of the shape functions $N_1$, we have:

$$B_{11} = \frac{\partial N_1}{\partial x} = \frac{\partial N_1}{\partial \xi}\frac{\partial \xi}{\partial x} + \frac{\partial N_1}{\partial \eta}\frac{\partial \eta}{\partial x} = -\frac{\partial \xi}{\partial x} - \frac{\partial \eta}{\partial x}$$

with:

$$\begin{aligned} \frac{\partial \xi}{\partial x} &= \frac{1}{|J|}\left(\frac{\partial y}{\partial \eta}\frac{\partial \xi}{\partial \xi} - \frac{\partial y}{\partial \xi}\frac{\partial \xi}{\partial \eta}\right) = \frac{1}{|J|}\frac{\partial y}{\partial \eta} = \frac{1}{|J|}\sum_i^n \frac{\partial N_i(\xi, \eta)}{\partial \eta} y_i \\ \frac{\partial \eta}{\partial x} &= \frac{1}{|J|}\left(\frac{\partial y}{\partial \eta}\frac{\partial \eta}{\partial \xi} - \frac{\partial y}{\partial \xi}\frac{\partial \eta}{\partial \eta}\right) = -\frac{1}{|J|}\frac{\partial y}{\partial \xi} = -\frac{1}{|J|}\sum_i^n \frac{\partial N_i(\xi, \eta)}{\partial \xi} y_i \end{aligned}$$

When dealing with first degree triangles, these terms are constant but, with higher order elements, they have to be evaluated at each Gauss point leading to the general form of the stiffness matrix:

$$K = \int\int B^T H B \, |J| \, t \; d\eta d\xi = \sum_j^{n_{gp}} w^j B(\xi^j, \eta^j)^T H^j B(\xi^j, \eta^j) \left|J(\xi^j, \eta^j)\right| t(\xi^j, \eta^j) \tag{4.5}$$

The superscript $j$ denotes the Gauss points, $n_{gp}$ the number of Gauss Points and $w^j$ the weights defined in the Gauss-Legendre quadrature rule [186].

### 4.1.2 Example of integration of a 1D linear extended finite element

The method that is used to compute the stiffness matrix is similar to the one presented in the previous section. However, because of the strong discontinuity of the shape functions that arises at the void-solid interface, one can observe that using a quadrature rule would lead to a totally unsatisfactory result even when increasing the number of Gauss points. If the discontinuity is not taken into account when the numerical integration is performed, this may lead to bad numerical results, and even to a non-invertible set of equations if the integration points 'miss' the discontinuity.

At first, one has to notice that because of the zero displacement field in the void domain, the void part of the element does not contribute to the stiffness matrix. Thus the integration procedure can be restricted to the solid sub-domain of the element (see gray domain in Fig. 4.2).

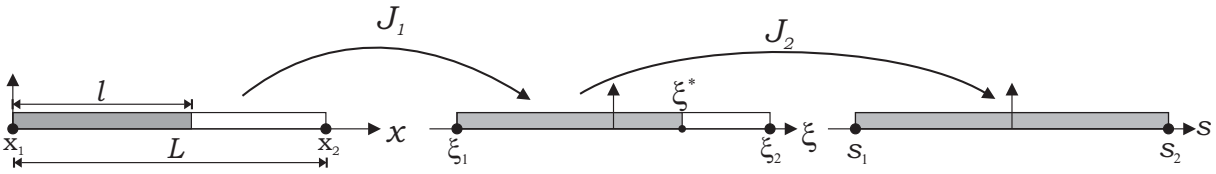



Figure 4.2: Rod element partially solid along a length $l$

For a linear 1D element, the shape functions can be defined in the $x$ referential of the rod as:

$$N_1(x) = 1 - \frac{x}{L}, \quad N_2(x) = \frac{x}{L}$$

The stiffness matrix is defined as:

$$\begin{aligned} K &= \int_0^l B^T H B \, dx = \int_0^l \begin{bmatrix} \frac{\partial N_1(x)}{\partial x} \\ \frac{\partial N_2(x)}{\partial x} \end{bmatrix} EA \begin{bmatrix} \frac{\partial N_1(x)}{\partial x} & \frac{\partial N_2(x)}{\partial x} \end{bmatrix} dx \\ &= \int_0^l \begin{bmatrix} -\frac{1}{L} \\ \frac{1}{L} \end{bmatrix} EA \begin{bmatrix} -\frac{1}{L} & \frac{1}{L} \end{bmatrix} dx = \frac{EAl}{L^2} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \end{aligned} \tag{4.6}$$

where $A$ is the cross section of the bar and $E$ the Young modulus.

Practically, this approach can not be used as we generally do not have the expression of the shape functions in the physical space. Instead, a reference space $\xi$ [-1,1] is introduced where the shape function are expressed:

$$N_1(\xi) = -\frac{1}{2}(1-\xi), \quad N_2(\xi) = \frac{1}{2}(1+\xi)$$

The mapping between spaces $x$ and $\xi$ is defined with the shape functions and the nodal positions of the element:

$$\begin{aligned} x &= \frac{1}{2}(1-\xi)x_1 + \frac{1}{2}(1+\xi)x_2 \\ &= \frac{1}{2}(1+\xi)x_2 = \frac{1}{2}(1+\xi)L \end{aligned}$$

From this relation, we obtain the position of the discontinuity $\xi^*$, the position of the node element in the reference element $\xi$, the bounds of integration in $\xi$ and the Jacobian of the transformation:

$$\begin{aligned} J_1 &= \frac{dx}{d\xi} = -\frac{1}{2}x_1 + \frac{1}{2}x_2 = \frac{L}{2} \\ 0 &= \frac{1}{2}(1+\xi)L \Rightarrow \xi_1 = -1 \\ l &= \frac{1}{2}(1+\xi)L \Rightarrow \xi^* = \frac{2l}{L} - 1 \end{aligned}$$

and rewrite (4.6) in terms of $\xi$:

$$\begin{aligned} K &= \int_{\xi_1}^{\xi^*} J_1^{-1} \begin{bmatrix} \frac{\partial N_1(\xi)}{\partial \xi} \\ \frac{\partial N_2(\xi)}{\partial \xi} \end{bmatrix} EA \begin{bmatrix} \frac{\partial N_1(\xi)}{\partial \xi} & \frac{\partial N_2(\xi)}{\partial \xi} \end{bmatrix} J_1^{-1} |J_1| \, d\xi \\ &= \int_{-1}^{\frac{2l}{L}-1} \frac{2}{L} \begin{bmatrix} -\frac{1}{2} \\ \frac{1}{2} \end{bmatrix} EA \begin{bmatrix} -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \frac{2}{L}\frac{L}{2} \, d\xi = \frac{EAl}{L^2} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \end{aligned}$$

However, the computation of the bounds of integrations in the $\xi$ referential in 2D and 3D is not straightforward. It is much more convenient to introduce a second mapping to obtain an integration domain with fixed bounds of integration and numerical scheme adapted to this domain. Introducing a second iso-parametric relation between $\xi$ and a new referential $s$ [-1,1] for the solid part of the element, one has:

$$\begin{aligned} \xi &= \frac{1}{2}(1-s)\xi_1 + \frac{1}{2}(1+s)\xi^* = -\frac{1}{2}(1-s) + \frac{1}{2}(1+s)(\frac{2l}{L} - 1) \\ \xi_1 &= -1 \Rightarrow s = -1 \\ \xi^* &= \frac{2l}{L} - 1 \Rightarrow s = 1 \end{aligned} \tag{4.7}$$

and the related Jacobian:

$$J_2 = \frac{d\xi}{ds} = -\frac{1}{2}\xi_1 + \frac{1}{2}\xi_2 = \frac{l}{L}$$

Transforming the equation (4.6) in terms of $s$, one obtains:

$$\begin{aligned} K &= \int_{-1}^{1} J_1^{-1} \begin{bmatrix} \frac{\partial N_1(\xi(s))}{\partial \xi} \\ \frac{\partial N_2(\xi(s))}{\partial \xi} \end{bmatrix} EA \begin{bmatrix} \frac{\partial N_1(\xi(s))}{\partial \xi} & \frac{\partial N_2(\xi(s))}{\partial \xi} \end{bmatrix} J_1^{-1} |J_1| \, |J_2| \, ds \\ &= \int_{-1}^{1} \frac{2}{L} \begin{bmatrix} -\frac{1}{2} \\ \frac{1}{2} \end{bmatrix} EA \begin{bmatrix} -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \frac{2}{L}\frac{L}{2}\frac{l}{L} \, ds = \frac{EAl}{L^2} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \end{aligned}$$

where $N_i(\xi(s))$ means that $N_i(\xi)$ is transformed into a function of '$s$' using (4.7).

Obviously, the three approaches present the same result. However, the last method is easier to make a systematic computer implementation as the integration schemes are generally defined for predefined reference element. In practice, the derivative of the shape functions are never expressed in terms of the variable $s$ because it is more convenient to use (4.7) to get the position of the Gauss points $s^j$ in the $\xi$ referential where the $B$ matrix is defined.

### 4.1.3 Integration of a 2D linear triangle extended finite element

In the case of a triangular element, we now have to deal with an integration domain that does not correspond anymore to the reference triangle in space $(\xi, \eta)$. Hence, one can see in Fig. 4.1 that the mapping of Gauss points with an unmodified integration scheme could bring the sampling points inside the void zone yielding to errors in the integration procedure or even worse, a structural matrix full of zeros.

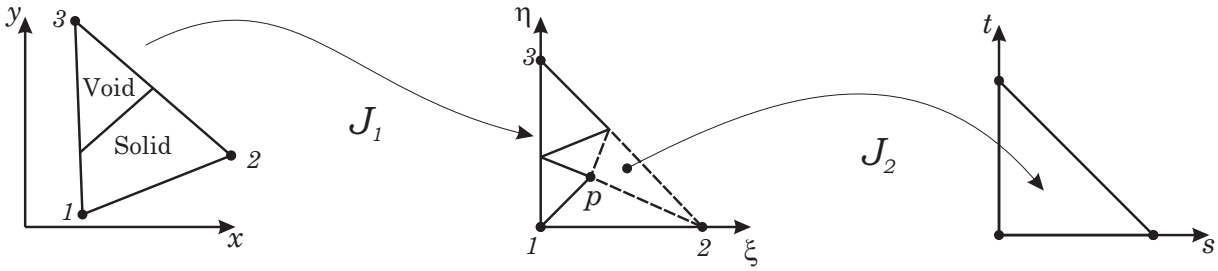



Figure 4.3: Triangle physical space, reference space and second reference space

The strategy is to divide the solid part of the original element into several sub-triangles (with the barycenter $P$ as one vertex for instance) conforming to the interface and the boundaries of the element itself. Then, a second reference space $(s,t)$ is introduced in which we successively map the sub-triangles of space $(\xi,\eta)$.

The classical FEM shape functions defined in this space $(\xi, \eta)$ are then integrated over each sub-triangles which cover the integration domain. Therefore, the stiffness matrix is now given by:

$$K = \int_{\Omega_{Solid}} B^T H B \, d\Omega_{Solid} = \sum_{\Delta_i} K_{\Delta_i} = \sum_{\Delta_i} \int_{\Delta_i} B^T H B \, |J_1| \left|J_{2_{\Delta_i}}\right| t \; dsdt$$

where $\Omega_{Solid}$ is the solid zone, $\Delta_i$ is a sub-triangle, $|J_1|$ and $\left|J_{2_{\Delta_i}}\right|$ are respectively the Jacobian of the transformation of $(x, y)$ space to $(\xi, \eta)$ and $(\xi, \eta)$ to $(s, t)$. Notice that the shape functions are still defined in the space $(\xi,\eta)$ whereas the integration domain is defined by the second mapping in the space $(s,t)$. No modification of the shape functions nor any enrichment is added to the approximation of the displacement field. This integral is finally computed with a Gauss scheme:

$$K_\Delta = \sum_{j}^{n_{gp}} B(\xi^j, \eta^j)^T H(\xi^j, \eta^j) B(\xi^j, \eta^j) \left|J_1(\xi^j, \eta^j)\right| \left|J_2(\xi^j, \eta^j)\right| w^j t(\xi^j, \eta^j)$$

Similarly, one can obtain for the following expression for the mass matrix:

$$M_\Delta = \sum_{j}^{n_{gp}} N(\xi^j, \eta^j)^T N(\xi^j, \eta^j) \left|J_1(\xi^j, \eta^j)\right| \left|J_2(\xi^j, \eta^j)\right| w^j \rho(\xi^j, \eta^j) t(\xi^j, \eta^j)$$

where $\rho$ is volumic mass. Finally, we can remark that a cut element (i.e. an element crossed by the discontinuity) will generally needs a number of Gauss points $n_{gp}$ which is higher than the number of Gauss Points used for a classical fully solid finite element. Hence, for the example of Figure 4.3, four Gauss points (one per sub triangle) are necessary to integrate the solid domain

if linear shape functions are used.

Remark that the sub-division depicted in the Figure 4.3 is not unique. For example, in order to minimize the computational effort of the stiffness matrix, one could divide the solid quadrangle in only two triangles with no visible difference in the result (in the case of linear triangle). The quadrangular solid region could also be mapped directly to a quadrangle. However, dividing the solid regions into triangles as detailed above is more systematic as it can be applied indifferently to triangular, quadrangular or pentagonal domains (see section 4.1.4).

In the case of the 2D linear triangles, the stiffness matrix can also be computed very efficiently with a minimum of Gauss points and no subdivision as follows:

- if the solid part is a triangular domain, the stiffness matrix of the element is computed with the method detailed before. Hence only one gauss point is necessary as the derivative of shape functions are constant on the domain of integration.
- if the solid part is a quadrangular domain, the stiffness matrix is equal to the stiffness matrix of the full element less the stiffness matrix of the triangular part. Here two computations of the stiffness matrix are realized with one Gauss point each.

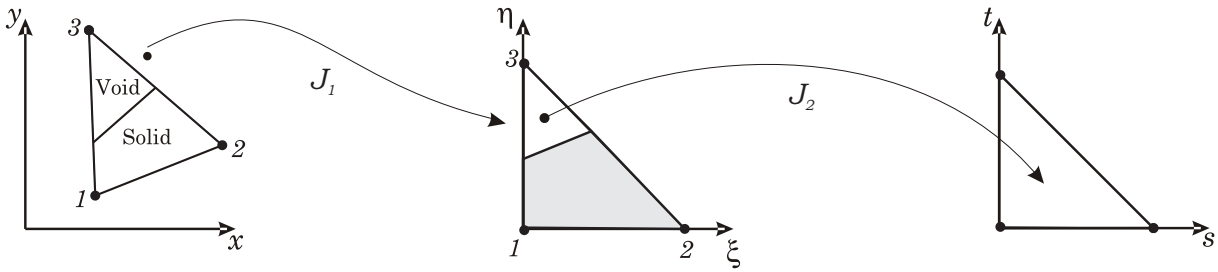



Figure 4.4: Alternative mapping for the linear triangle

Finally, we can add also that when using first degree triangular extended finite elements, the stiffness matrix can also be computed very efficiently with only one gauss point indifferently of the topology of the element by remarking that (as in classical finite element method) the stiffness matrix is independent of the geometry of the material domain. Hence, as the derivative of the shape functions are constant on the domain $d\Omega_{Solid}$, the expression of the stiffness matrix can be reduced to:

$$K = \int_{\Omega_{Solid}} B^T H B \, d\Omega_{Solid} = B^T H B \int_{\Omega_{Solid}} d\Omega_{Solid} = K_\Omega \frac{\int_{\Omega_{Solid}} d\Omega_{Solid}}{\int_\Omega d\Omega} = K_\Omega \frac{V_{solid}}{V_{full}}$$

where $K_\Omega$ corresponds to the stiffness matrix of the corresponding standard finite element. The stiffness matrix is equal to the stiffness matrix of the full parent element multiplied by the ratio of material volume ($V_{solid}$) over the elementary area ($V_{full}$). These two remarks hold also for first degree tetrahedron and linear rod elements.

### 4.1.4 Integration of a 2D linear quadrangle X-FEM

The principle of the method presented for the triangular element still holds for the quadrangle. We only have to deal with non constant Jacobian $J_1$ and matrix $B$ inside the domain since

the shape functions are now bilinear. In the case of bilinear shape functions or more generally speaking with non linear shape functions, the exact interpolation of the interface is in general curved (see Fig. 4.5). But, due to the difficulty to obtain this curved interface in practice, the interface is generally considered as a straight line joining the points of intersection of the interface and the element edges. An example of the mapping sequence is sketched in the Figures 4.5 where the dashed line represent the linearly interpolated interface.

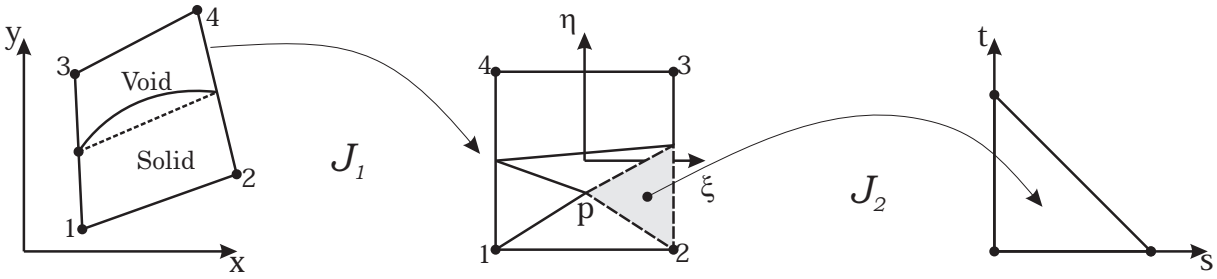



Figure 4.5: Quadrangle physical space, reference space and second reference space

The shape of the solid part solely influences the division into sub-triangles. When the solid zone is a pentagon, one has to sum over five sub-triangles, and four when the solid zone is a quadrangle and three when dealing with a triangular filled region. The sub-division is not unique and some authors generally split a pentagon into three triangles while we chose to split into five triangles around the barycenter of the material domain.

Another technique consists in decomposing the initial quadrilateral element into two triangles and then interpolate linearly the interface on these triangles. As it can be seen in Fig. 4.6, this technique generally enables a better interpolation of the interface.

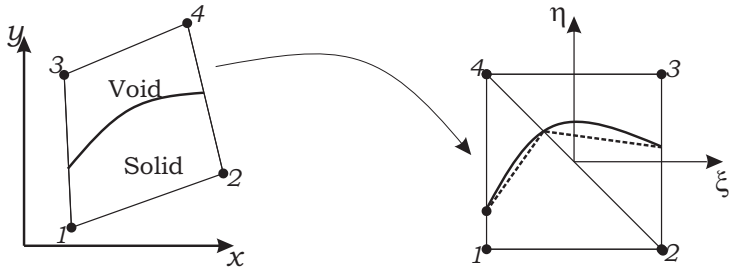
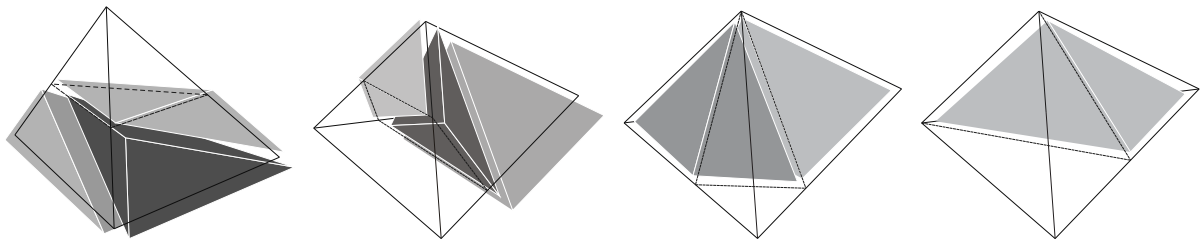



Figure 4.6: Decomposition of the quadrangle into two triangles

### 4.1.5 Integration of 3D X-FEM elements

With 3D elements, the method is extended to the third dimension and the sub-division of an element yields several sub-tetrahedral domains. The Figures 4.7 illustrates the 4 possible cases of the intersection of a plane with a tetrahedron and the sub-division of the solid part into integration cells (the positive nodes lie in the material whereas negative nodes are in the void part).

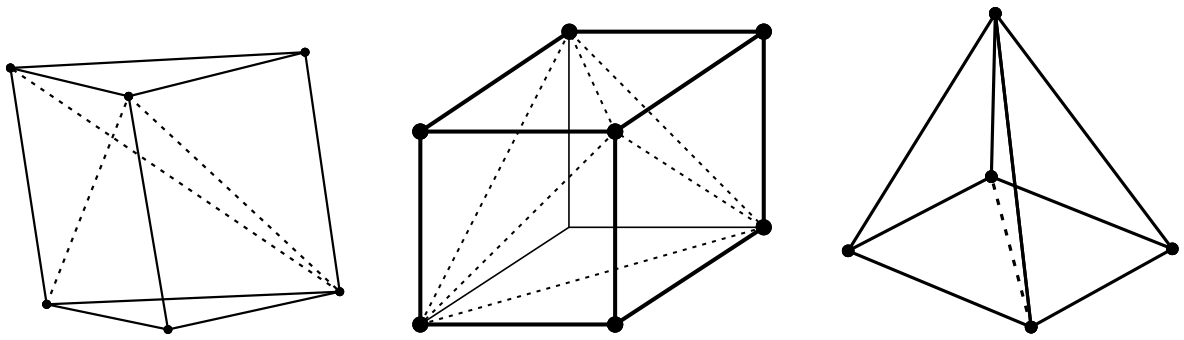
Here also, the interface can be curved inside the reference element in case of non linear shape functions (in hexahedron for instance) if it is exactly interpolated. However, following the method



(a) One negative node (b) Two negative nodes (c) One negative node and one zero node (d) One negative node and two zero nodes

Figure 4.7: Intersection of a plane and a tetrahedron and sub-division

applied with the quadrilateral element, the interface is considered as a plane inside the reference element. Then, for all other topologies of elementary finite elements: hexahedron, wedges and pyramids, the intersection of the interface with the element is never computed in practice. Instead, the initial element is firstly decomposed into tetrahedron cells, and each of these tetrahedrons is then divided into integration cells conforming to the iso-zero level set surface (see Fig. 4.8). Doing so, the level set surface is interpolated linearly inside each tetrahedral cell without using the parent element shape functions. One can therefore note that even if the interface is a straight line on the element faces of the hexahedron, the interface can be non planar and generally yield to multiple planar surface interfaces. On one hand, the direct sub-division of the element could generally reduce the number of integration cells and so, the number of integration points but, on the other hand, the tetrahedralization permits to limit the complexity of the computer code as the subdivision routines of a tetrahedron is re-used for other elementary topologies. Moreover, it allows a good compromise between a strictly plane interface representation inside the element and a curved interface.



(a) Wedge to 3 tetrahedron (b) Hexaedron to 5 tetrahedron (c) Pyramid to 2 tetrahedron

Figure 4.8: Tetrahedralization of wedge, hexeadra and pyramid

### 4.1.6 Bi-material

When modeling material interfaces or discontinuous fields and non-smooth fields with discontinuous gradient, the integration method presented above still holds. The only modification that we need to perform is to enlarge the matrix $B$ as we now have to deal with additional shape

functions and we have to integrate over the two sides of the discontinuity boundary. The general form of the $B$ matrix becomes (in the case of a linear triangle):

$$B = \begin{bmatrix} B_{11} & 0 & B_{12} & 0 & B_{13} & 0 & B_{11}^* & 0 & B_{12}^* & 0 & B_{13}^* & 0 & \\ 0 & B_{21} & 0 & B_{22} & 0 & B_{23} & 0 & B_{21}^* & 0 & B_{22}^* & 0 & B_{23}^* & \ddots \\ B_{21} & B_{11} & B_{22} & B_{12} & B_{23} & B_{13} & B_{21}^* & B_{11}^* & B_{22}^* & B_{12}^* & B_{23}^* & B_{13}^* & \end{bmatrix}$$

where $B_{ij}^*$ is the derivative of the enriched shape functions. The order of integration depends on the nature of these enriched shape functions and has to be adapted to obtain an accurate numerical integration. Moreover, the order of integration does not depend only on the shape function order because we have now to take into account that the maximum degree of the function to integrate is equal to $p + n - 1$ where $p$ is the degree of the classical shape functions and $n$ is the degree of the enrichment functions. However, in some cases, the enrichment is not polynomial and a new integration scheme taking into account the non-smooth or the singular behavior of the approximation have been proposed [12, 16, 96].

For the sake of completeness, we should also mention that various authors have proposed different integration methods to circumvent the element subdivision technique and the two mappings. Some of these methods are directly based on the observation of the enrichment function characteristics while the others are based on mapping adaptation or integral refactoring (see [31, 85, 175]). Bordas *et al.* [126] have proposed to apply the smoothed finite element method which consists in replacing the domain integral by a contour integral along the boundary of the element to avoid the requirement of an iso-parametric mapping. This approach leads to a smoothed X-FEM approach. Finally, Natarajan *et al.* [125] proposed special integration schemes for *n-sided* polygons that eliminate the element sub-division and the two successive mappings by using Schwarz-Christoffel conformal mapping on unit disk. Recently, Mousavi *et al.* [122] also proposed a numerical method to determine the position of the Gauss points in arbitrary $n$-sided polygons. This method is very interesting but unfortunately it is currently available for 2D elements only.

### 4.1.7 High order extended finite elements

Like in FEM, it can be useful to resort to high order approximations to obtain more accurate solutions, higher rates of convergence and an enhanced geometrical representation of the iso-zero level set. Formally, the generalization of X-FEM to high order is 'in principle' straightforward. However, according to Ref. [98, 161, 185] considering high order X-FEM, the standard X-FEM decomposition (polygonal and polyhedral) providing a rectilinear interface approximation generally leads to sub optimal rates of convergence. Only a better absolute value of the error is obtained when high order shape functions are combined with a first order geometrical interpolation. Moreover, the introduction of high order shape functions requires careful attention to the behavior of the enrichment functions and the blending elements. Actually, we can classify two different methods to implement high order X-FEM. If one uses a high order FEM approximation for the physical fields, the same approximations can be used to represent the iso-zero level set and it is possible to represent curved interface inside elements. Mayer *et al.* [108] presented an application of high order interface element subdivision for X-FEM fluid structure interaction, while Legay *et al.* [98] has developed a subelement integration technique considering the interface curvature with 6 and 10 node triangles for quadratic and cubic elements respectively. More recently, an optimal rate of convergence has been reported in Cheng and Fries [43] by using hybrid 5-nodes triangular and 6-nodes quadrangular subelements. As it can be seen in Figure 4.9, these elements present only a single curved edge supporting 4 nodes that is mapped onto the

curved interface. They found that placing 4 nodes on the curved interface is sufficient to ensure optimal rates of convergence for holes and bi-material X-FEM formulations using quadratic and cubic elements with the quadrature distribution represented in Figure 4.9 (crosses). Presently, this method has not been yet extended to 3D elements.

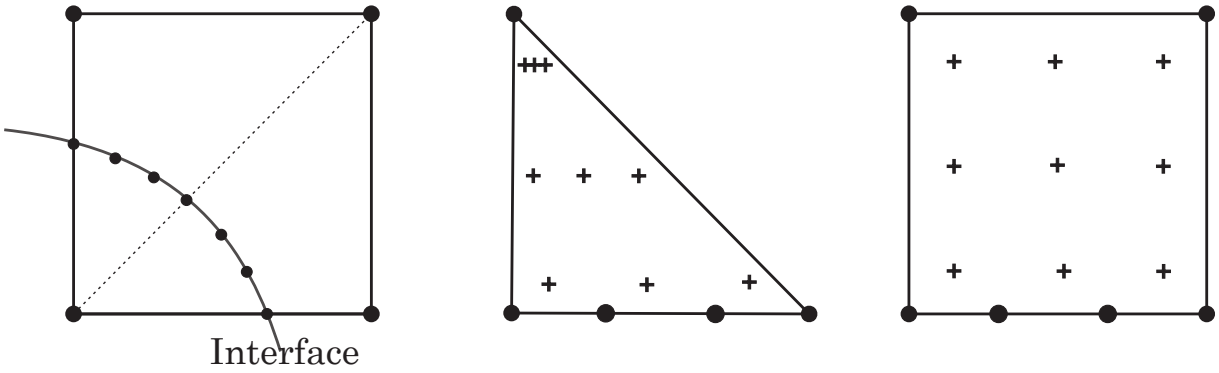



Figure 4.9: Subcell division and integration with Cheng and Fries method [43]

On the other hand, the element can be recursively divided into subelements using a predefined division pattern. Inside each subelement, the interface is interpolated linearly from the nodal Level Set values of the initial element and we obtain a piecewise linear (or piecewise planar in 3D) interface inside the initial elements (see Fig. 4.10 and 4.11). The number of integration subelements is then higher than with a linear element and the position of the integration points is adapted to the order of the function to integrate. Compared to the method of Cheng and Fries [43], the number of integration points is much higher as we have to deal with multiple subelements, and the geometrical accuracy is lower as the interface remains linear.

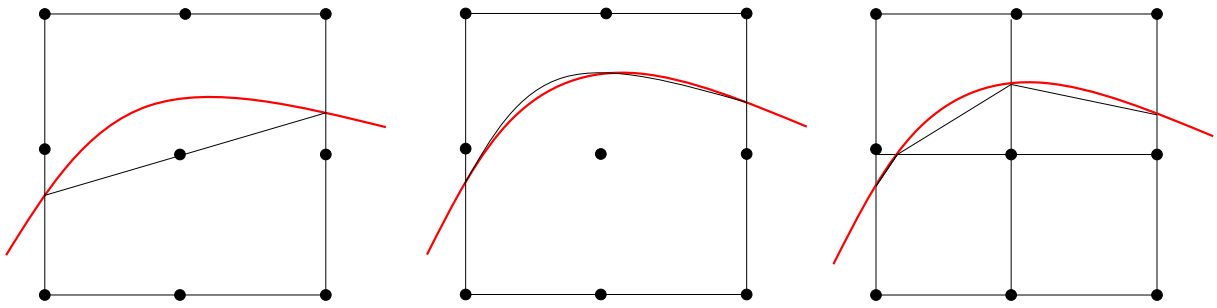
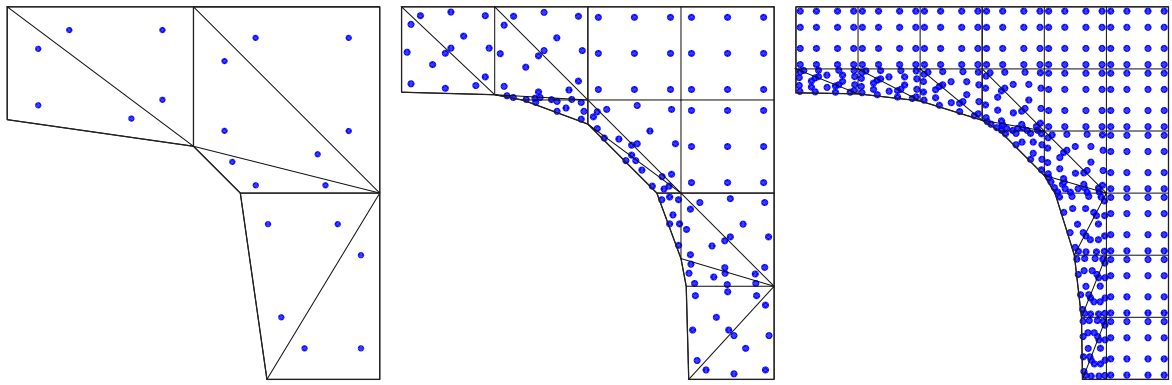


Figure 4.10: Recursive subdivision and interface interpolation on biquadratic element using linear interpolation, quadratic interpolation and recursive subdivision

However, as we can notice in Figure 4.11, the geometrical accuracy is already very good at second order while small difference can be observed with third order. Moreover, the algorithm can be implemented without difficulties to 3 dimensions and the extension to higher order is straightforward. In OOFELIE, we have chosen to use a recursive sub-division for the high order elements. According to Dréau *et al.* [53] who studied the effect of recursive subdivisions within X-FEM, we obtained quasi-optimal rates of convergence with this sub-division technique as we

will see in section 4.4.1.



(a) First order / Volume relative error 25%

(b) Second order / Volume relative error 4%

(c) Third order / Volume relative error 3%

Figure 4.11: Recursive subdivision and integration points on 4 elements in OOFELIE

## 4.2 Implementation in OOFELIE

Compared to a FEM code, there are some major aspects that need to be modified in order to develop an X-FEM code. First of all, the element subdivision has to be implemented. This task can rely on an explicit geometrical description of the interface but it is greatly improved by using a Level Set description. Then according to the integration method presented before, the quadrature procedure has to be adapted to handle the integration of the various enrichment functions and/or the solid material part of the element. Depending on the considered type of discontinuity, the enrichment functions have to be implemented. Finally, the code should be able to deal with a variable number of degrees of freedom per node (that should ideally be dynamic) and an adaptive integration rule per element. The post processing tools (i.e. visualization tools) need also to be extended to deal with results that may hold inner element discontinuities. All these steps are detailed in the following.

### 4.2.1 Interface description and element subdivision

At the dawn of the extended finite element method, the interfaces were modeled explicitly by polygons, polyhedrons and polylines for cracks. So, the slicing and the subdivision of the elements was carried out in the physical space $(x, y, z)$ and each subcell mapped from this space to the reference space. While this 'inverse' mapping is linear for triangular/tetrahedral linear element and quadrangle/hexahedron with parallel edge/face it turns out that, in the general case, a set of a non linear equations has to be solved. With high order shape functions, this mapping can become time consuming and also brings some accuracy difficulties.

Rapidly, researchers have adapted the representation of interfaces with the Level Set Method within the X-FEM. The main reasons are that the Level Set description facilitates several problems. The first advantage of the Level Set over an explicit representation of an interface is to simplify the search of the intersection of this interface with element mesh. While an explicit geometrical representation would need a quadtree/octree tool to efficiently detect the set of

'probably cut element', the Level Set method gives a convenient way to detect these elements.

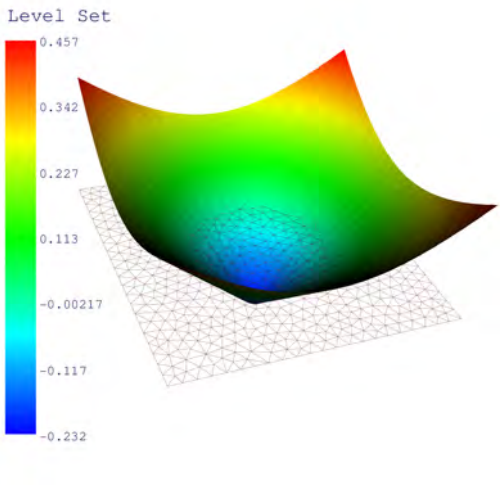



Figure 4.12: Representation of the Level Set describing a circular interface

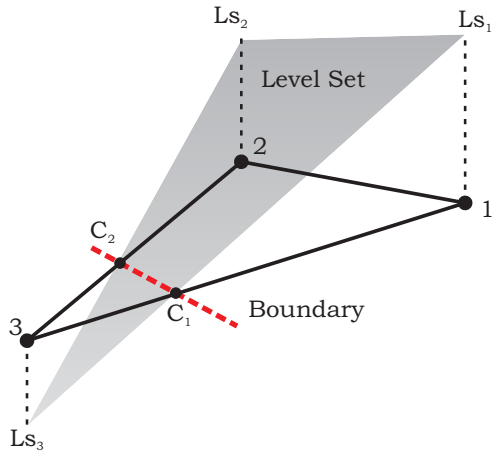
When dealing with one Level Set as in Figure 4.12, the classification of elements according to their nodal Level Set values can be carried out directly and the selection of intersected element is straightforward. It consists in the set of elements that support both negative and positive Level Set nodal values. On these elements, each edge that possesses negative (or zero) and positive values are intersected by the iso-zero Level Set surface and they provide the intersection points $C_1$ and $C_2$ (see Fig. 4.13 (a)). Given these points and the element nodes, two domains are obtained on which the numerical integration should be carried out. According to the Level Set sign or identifier, material properties are assigned and enriched shape functions can be defined.

While this procedure sounds to be very easy, it is in practice more complex as we have to deal with different patterns of element subdivision (10 different patterns for a triangular element). Moreover, this subdivision is dependent of the configuration of the interface inside the element (i.e. element numbering) and one can imagine that the task is also more complex when dealing with 3D elements. Furthermore, to prescribe Dirichlet or Neumann boundary conditions on the interface, disposing of valid finite element mesh of the iso-zero Level Set is helpful. Hence, one should pay attention to create only unique entities, consistent with inter-element relations. For instance, in Figure 4.13 (a), the points $C_1$ and $C_2$ have to be unique and are shared with adjacent elements. When the model has more than one Level Set, the detection of cut elements is more complex. Each Level Set nodal values has to be stored and the mesh slicing should be processed recursively.

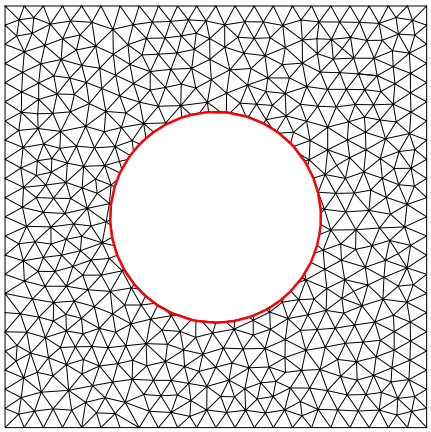
### 4.2.2 Integration

In the case of material-void interfaces, elements are of only three types: solid ones, void ones and intersected ones.

The first type of elements are such that all the nodes lie fully inside the void. These elements (white elements in Fig. 4.14 (a)) present negative nodal Level Set values for all nodes, have no




(a) Representation of the Level Set inside an element



(b) Representation of the model with cut and filled elements only

Figure 4.13: Geometry description and sliced mesh

contribution and they are consequently removed from the problem (their DOF are not assembled in the structural matrices).

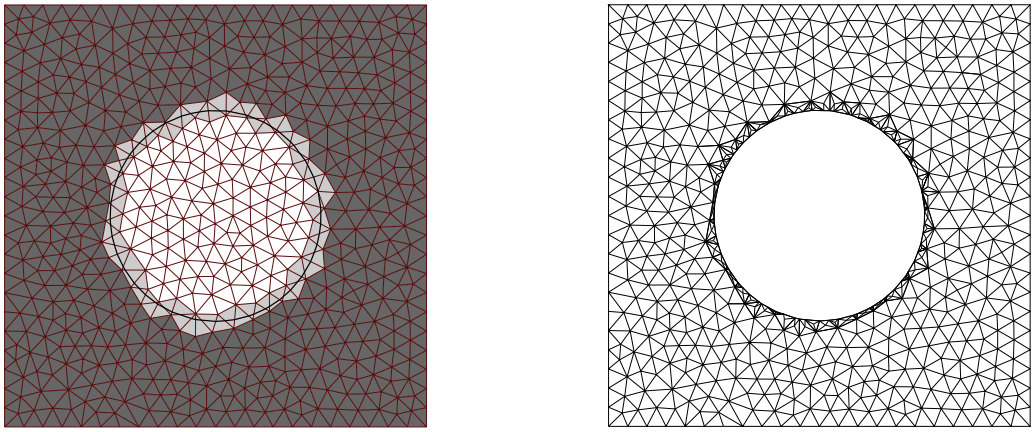
The filled elements (dark gray ones in Fig. 4.14(a)) have all their nodes lying in the solid domain. They are treated as classical FEM elements and do not need special attention.

The partially filled elements (light gray ones in Fig. 4.14(a)) crossed by the interface have a mix of void and solid nodes. They require an X-FEM integration procedure of the elementary structural matrices.

In the case of cracks or bi-material interfaces, the surrounding band of elements sharing a node with the intersected elements requires also a special processing since they may support extended degrees of freedom and enriched shape functions. Depending on the character of the enrichment shape functions, the Level Set sign may not be sufficient to classify the elements and detect which element should be enriched. In this case, an element is enriched if one of its nodal shape function support is cut by the discontinuity. This remark holds for blending elements as well.

From the sub-division obtained by the mesh slicing, two integration domains (for one single Level Set) conforming to the discontinuity are obtained and the Gauss points are placed within these domains according to the quadrature rules. Inside OOFELIE, different strategies have been implemented to create the integration domains. For instance, when cutting a quadrangular element, one can choose three strategies which consist in:

- An initial subdivision of the quadrangle into two triangles, all quadrangles created are then divided again into triangles;
- No initial subdivision of quadrangles but all quadrangles or pentagons are divided into four or five triangles after inserting of a new point at barycenter;
- No initial subdivision of quadrangles but all quadrangles or pentagons are divided into triangles according to a criterion of best geometrical quality.



(a) Representation of the different element type

(b) Representation of the model with the sub-triangles

Figure 4.14: Cut mesh and sub-triangles for integration

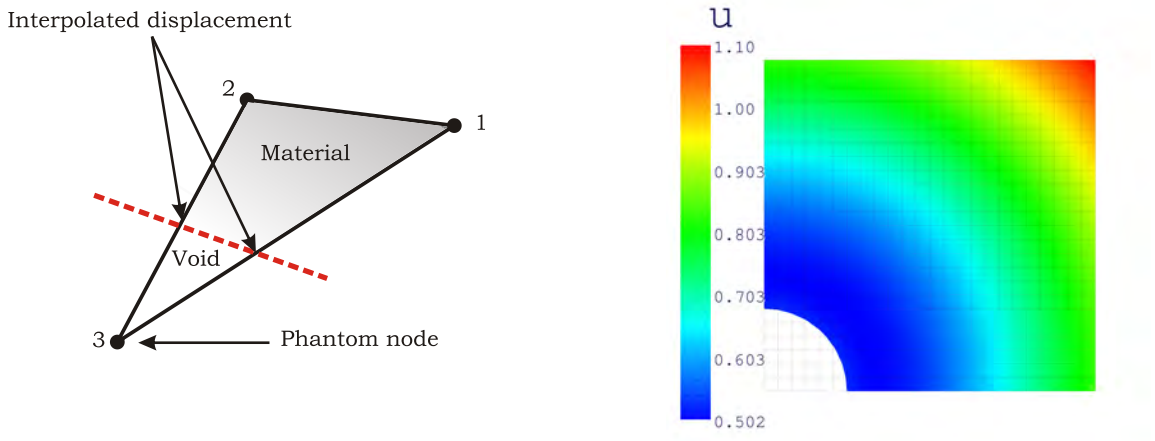
Similar rules exist for three dimensional problems where we can choose to have a fast slicing strategy or to select the best sub-division according to geometrical criteria.

After the element subdivision, the following mesh of Figure 4.14 (b) is obtained. One has to notice that the sub-triangles are only a background mesh used for the integration and that the size of the problem is not augmented regarding to a classical finite element model. The structural matrices of these elements are computed with the procedure detailed before. However, to obtain an implementation that is very close to a classical FEM implementation, the sub element mapping and the second Jacobian $J_2$ is invisible for the user. First, the subelement obtained from the mesh clipping are stored using a hierarchical father-children relation. The father being the original finite element and each child element being a sub-domain of integration characterized by a Jacobian ($J_2$) and a set of material properties associated to its Level Set identifier. Then, given these relations, the children can provide the set of Gauss point positions and the related weights $(w^{j^*}) = w^j J_2$) in order to proceed to the numerical integration transparently as a classical FEM:

$$K = \sum_{j}^{n_{gp}} B(\xi^j, \eta^j)^T H(\xi^j, \eta^j) B(\xi^j, \eta^j) w^{j^*} \left| J_1(\xi^j, \eta^j) \right|$$

### 4.2.3 Visualization

As most processing tools are not able to deal with inner element discontinuities, we have extended the visualization tools of OOFELIE to handle X-FEM elements. For instance, with material-void elements, one obtains nodal values for the elements that are present in the stiffness matrix. However, as the cut elements support *phantom* nodes (i.e. nodes lying in the void part), these nodes also hold a nodal displacement. The iso-zero Level Set boundary value is obtained from a shape function interpolation based on the nodal values (see Fig. 4.15).




(a) Representation of the displacements for triangular X-FEM elements

(b) Representation of the displacements for quadrangular X-FEM elements

Figure 4.15: Representation of the displacements

### 4.2.4 Capabilities of the present implementation in OOFELIE

Actually, we have introduced 2D and 3D elements of different types (triangle, quadrangle, tetrahedron, pyramid, hexahedron and prism) with several approximation orders (up to third degree). These extended finite elements are able to treat both material-void interfaces and bi-material interfaces. Beside these 'classical FEM elements', axisymmetric elements have been introduced.

As mentioned before, the choice of using the extended finite element method for shape optimization has been motivated by the ability of the method to work with a fixed mesh during structural shape evolution. It is preferable that the computational cost added by the mesh slicing and element integration should be lower than the computational effort involved by resorting to a classical shape optimization relying on mesh adaptation and remeshing. Therefore, in our development, a lot of attention has been devoted to obtain a computer code that could, reliably and efficiently, cut the mesh. As explained later in the optimization section, the sensitivity computation and the shape evolution will call for numerous cuts of the mesh. Hence, ideally, the penalty of managing non-conforming FE mesh should be less time consuming than a complete remeshing of the model.

## 4.3 Numerical accuracy

### 4.3.1 Numerical integration of the structural matrices

This section is devoted to explain and fix two kinds of numerical problems that might occur with the X-FEM for material-void interfaces when solving real life applications. The first one happens when the remaining solid part of the element is a quadrangle that is nearly degenerated into a triangle. The case is illustrated in Fig. 4.16. This quadrangle can be divided into four triangles, but one of these triangles is ill-conditioned (nearly flat triangle) and may introduce singular Jacobian matrices. In practice, it is not really a problem, because one can simply ignore

this very flat triangle (ABE in Fig. 4.16) and consider only the three sub-triangles to proceed to the numerical integration. One can also consider the quadrangle solid domain as a triangle and divide it into three triangles.

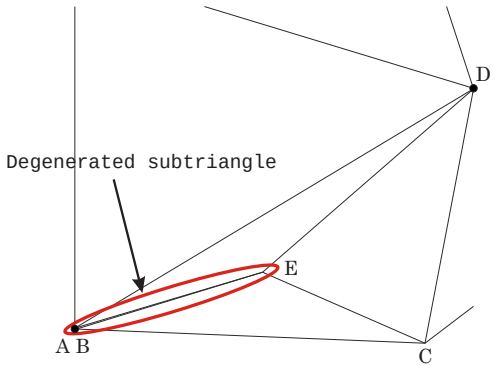
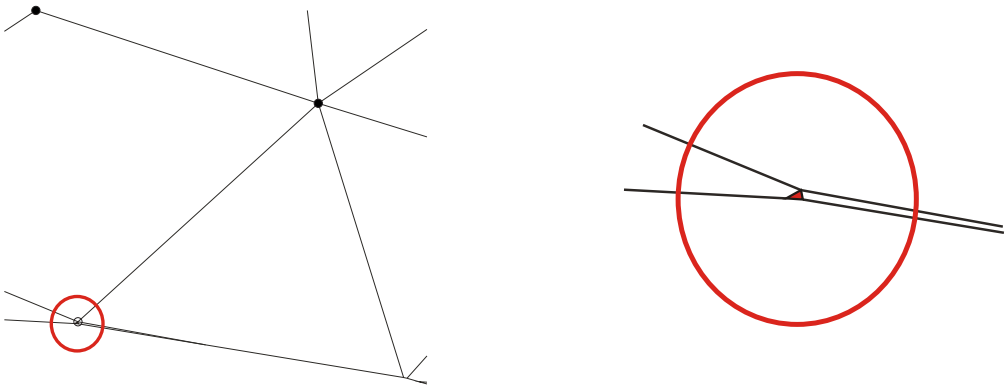



Figure 4.16: Quadrangle degenerated into a triangle

The second kind of numerical error occurs when the area of the solid part becomes very small regarding to the area of the supporting element (Fig. 4.17). In this case, with a material-void approach, depending on the size of this solid part, the structural matrices can become ill conditioned due to the presence of near-zero values on the diagonal. A similar problem can occur in the same situation with a Heaviside step enrichment. If the support of a given node is cut by a crack and the size of one part is very small compared to the other one, the step function is constant on nearly all the support. This leads to the introduction of an enrichment that is close to a classical shape functions and so it introduces an ill-conditioned structural matrix due to linear dependencies.



(a) Small solid triangle (b) Zoom of the small solid triangle

Figure 4.17: Illustration of a small solid element leading to rank deficiency of stiffness matrix

To solve this problem, some authors (see for example Ref. [34, 51]) suggest not to include these elements and eliminate them from the model when building the stiffness matrix.

### 4.3.2 Estimation of stresses

The computation of the stresses or the elementary energy do not call for any special procedure in X-FEM. Indeed, in the non-conforming elements, the strain are computed at the Gauss points used to compute the stiffness matrix and the displacements defined on the global element. Hence, displacements defined at *phantom nodes* are used and the stresses and the strain energy are classically given by:

$$\sigma_e = HB_e u_e, \quad W_e = \frac{1}{2} u_e^T K_e u_e$$

where $u_e$ is the element displacements vector, $K_e$ and $B_e$ are, respectively, the element stiffness matrix and the shape functions derivate matrix.

However we could observe that the very small solid elements are the source of an important overestimation of the local stresses. To illustrate the difficulty we consider the problem of a quarter plate with a hole under a uni-axial load along the $x$ axis (see Fig. 4.18). It shows the stress component $\sigma_x$ (computed at Gauss points) obtained with a FEM model, whereas the right Figure 4.19 illustrates the same results with an X-FEM analysis.

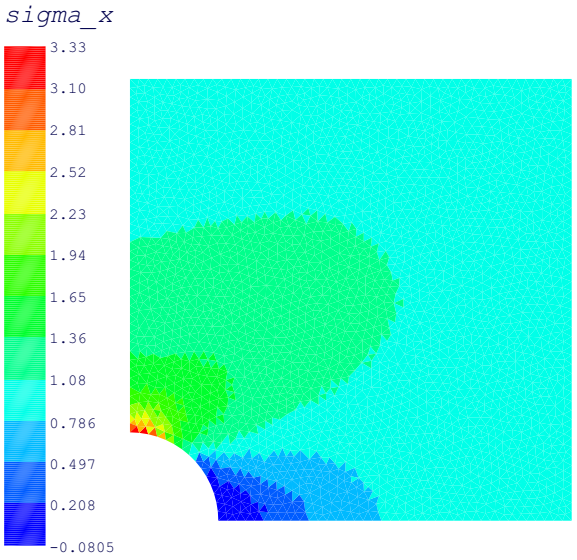



Figure 4.18: FEM stresses distribution

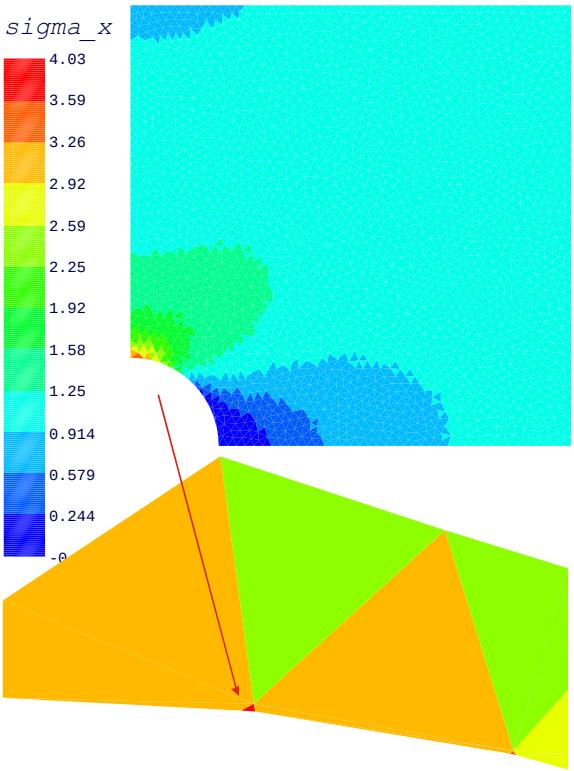
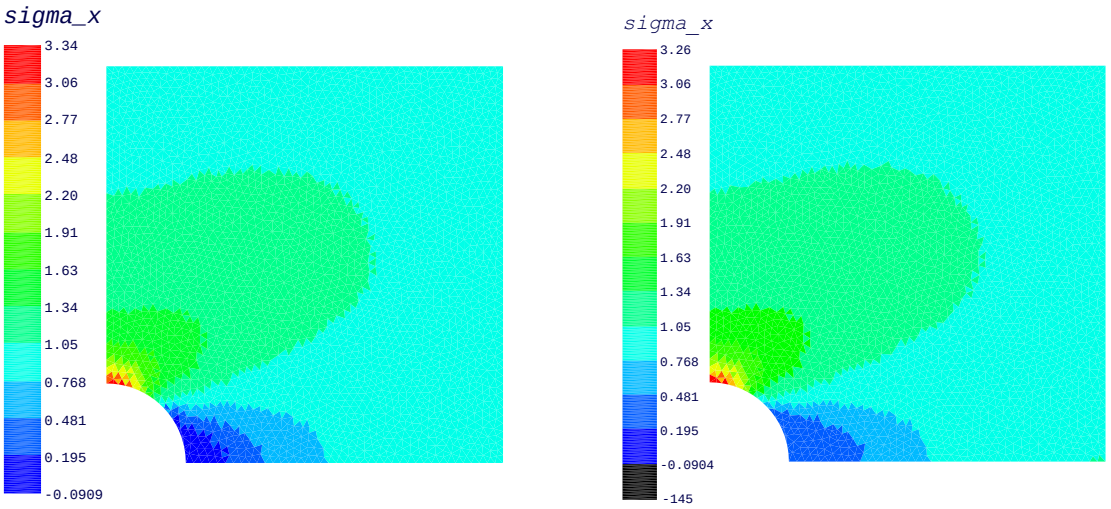



Figure 4.19: X-FEM stresses distribution

From Figure 4.19, it appears that a large error of roughly 20% occurs on the maximum stress for the X-FEM model (the same occurs for the $\sigma_y$ component with loading along the $y$ axis). This overestimation of the stress level is due to a small solid element as pointed out in the zoomed part of Fig. 4.19. If the boundary of the circle is slightly modified in order to remove the small cut element, the agreement between the results obtained with FEM and X-FEM become excellent (Fig. 4.20 (a)), which demonstrates that the small element was responsible of the error.




(a) X-FEM stresses estimation without small solid elements for a modified radius $r(1+10^{-4})$

(b) FEM analysis on the X-FEM mesh

Figure 4.20: X-FEM stresses on modified geometry and FEM Stresses on X-FEM mesh.

To identify the source of the problem, we have realized a first FEM analysis on the X-FEM mesh, replacing quadrangular intersected elements with quadrangle elements to see wether this problem also appears with a FEM model. In Figure 4.20 (b), we can clearly notice that the errors is even higher than with a X-FEM. A zoom on the highly stressed element (Fig. 4.21) shows that this element is overstretched and completely flipped thus introducing very high stress peak in the model.

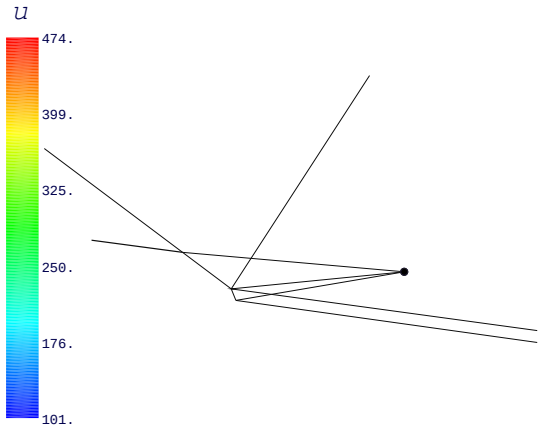



Figure 4.21: Strain of the FEM small element

Due to its size, this element has a very low rigidity compared to the neighboring elements and its deformation is imposed by its neighbors. Furthermore, the zone where the element is located presents a high gradient such that it is highly distorted and exhibits a high stress level. Hence,

it can also be shown that if a model presents a very small element in a zone where the strain is constant or moderate, no stress peak occurs.
In practice, we can not modify globally the boundary with a small perturbation as illustrated before. Furthermore there is no guarantee that the new interface is not going to introduce any other very small elements somewhere else in the model. It is therefore clear that a local modification or a post processing of the stresses must be preferred. Thus, to prevent the phenomenon, one can resort to the following strategies:

1. Eliminate from the model the elements that have a too small solid part when building the stiffness matrix. This criterion, can be based on a volume aspect ratio of the filled part to the global element, has been presented in Ref. [51];

2. Post process the stress results and eliminate the stresses when the ratio of the solid area to the supporting area is too small;

3. Use a smoothing stress method;

4. Add additional stiffness to the small element using for instance springs between the involved degrees of freedom;

5. Add a small local perturbation to the boundary on the node in order to remove the small elements (i.e. set the nodal Level Set value to zero). This procedure, called $fit\ to\ vertex$, has been proposed in Ref. [113].

6. Move the nodes to the material domain or to the void part.

7. Fill the void zone with a very soft material as SIMP material in topology optimization.

The first method, which consists in eliminating the small solid element, can be employed without any difficulties and is very similar to method 2. The difference being that the elementary stiffness of the small elements are readily suppressed from the global stiffness matrix in the method 1 whereas the rigidity of the element is kept in solution 2. These solutions are fairly simple since they just call for a post-processing of the results or a pre-processing of the element before assembling.

The choice 3, using a stress smoothing method, can also be considered as related to the method 2 but in this case, the smoothing procedure do not cut-off the peak but rather smooth out the stresses according to a smoothing criterion. It is investigated in section 4.3.4.

The method 4 aims at introducing some extra rigidity on the DOF of the small elements to prevent them from being over-strained. However, this method is not very appealing as it raises the question of where to introduce the springs and how much stiffness should be introduced.

The method 5, '$fit\ to\ vertex$', has been proposed by Moës *et al.* in [113] and generally performs well. In practice, this can be implemented by pre-processing the Level Set and assigning a zero nodal value for nodes very close to the boundary. The criterion used to detect close nodes being based on a ratio of solid and void elementary edge length or Level Set values equivalently. While this method is generally sufficient to avoid peak stresses, we have observed that the '$fit\ to\ vertex$' strategy is the source of errors when computing sensitivities in optimization problems involving geometrical dependent loads. Hence, as these loads depend on the interface geometry, this method cannot be applied as it may modify the sensitivity analysis. Moreover, this strategy

can sometimes fails if the ratio chosen in the criteria is too small.

The choice of method 6, 'move node to solid or void domain', is different from modifying the boundary (choice 5) as in this case we relocate all the nodes near the boundary to obtain a better ratio between void and filled part. We investigate this approach in section 4.3.5.

Finally, one can also imagine to fill the void domain with some soft material in order to add some stiffness to the cut elements as in the solid material with penalization. This can be achieved with a material-void approach or with a bi-material enriched approximation. It is investigated in section 4.3.3.

Some of these strategies are presented and compared in the following of this section.

### 4.3.3 Replacing void by a soft material

In this paragraph, we investigate the influence of considering the void as a soft material. To this end, the extended void-material elements are replaced by bi-material elements enriched with the ridge function (see (3.5)). Hence, the cut elements are composed of the original material properties on one side and of a very soft material instead of void. The elements fully located in the void are assigned to a soft material while the remaining elements are kept as normal ones. The soft material properties should preferably be as low as possible to reduce the perturbation introduced by the approximation of the void by a soft material. In Fig. (4.22) top, we plot the evolution of the maximum stress (with respect to FEM solution), maximum displacement and strain energy versus the Young modulus ratio between the solid material and the soft one. Figure 4.22 shows that when this ratio tends to zero, the X-FEM solution converges to the FEM solution and this suggests to choose the lowest ratio. However, as depicted on the lower graph, we can see the relative conditioning number of the stiffness matrix is deteriorating when the soft material tends to be softer. Thus, a good compromise is to limit the soft material to a ratio of $10^{-3}$ compared to the real material as we already obtain a good solution and do not deteriorate too much the conditioning number of structural matrices. Note that in topology optimization, the usual ratio between soft and real material is about $10^{-9}$.

The Figure 4.23 illustrates that the introduction of a soft material performs very well as it completely suppresses the overestimated stress with a Young Modulus ratio of $10^{-3}$. Notice that the restitution of an acceptable stress level is not due to the introduction of a soft material. The additional degrees of freedom provide enriched kinematic space preventing the element to be over strained. Indeed, integrating both sides of the element with a soft material in place of the void and no enrichment does not reduce the stress level.

However, this method presents the drawback of introducing more elements than necessary, we have to deal with extra degrees of freedom for the enrichment, which can be a problem for the sensitivity analysis if the interface crosses new elements when perturbing the interface. The introduction of a soft material may also be the source of the local-mode phenomenon that appears in topology optimization [136].

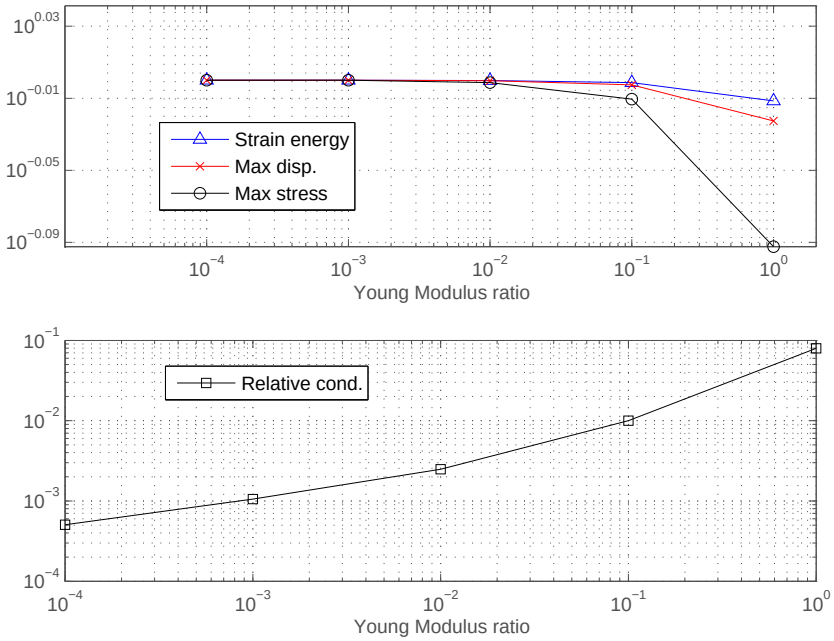



Figure 4.22: Convergence of the strain energy, max. displacement and max. stress and of the condition number of stiffness matrix with respect to the Young modulus ratio.

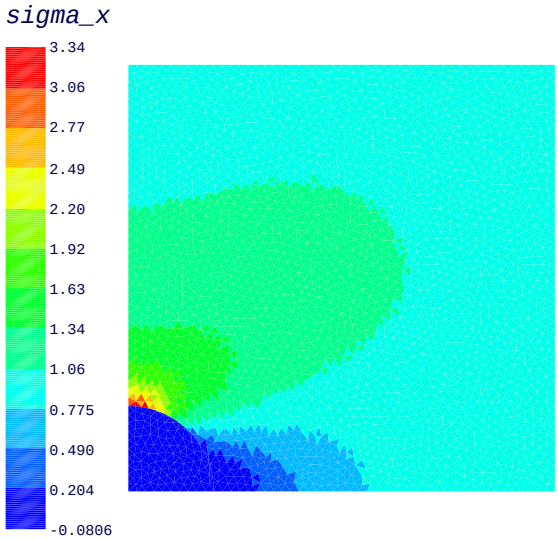



Figure 4.23: X-FEM stresses with a Young Modulus ratio of $10^{-3}$ for soft domain

### 4.3.4 Stress smoothing

#### Averaged stress

Another option consists in smoothing or filtering the stresses. We first made an attempt to compute a nodal stress by averaging the stress of the elements connected to the nodes. While this approach can generally reduce the overestimated peak stress, if the highest stress appears in an element located at a corner, the peak is not smoothed as no element is connected to the corner node. Thus, this approach is not really appropriate. The second attempt is to smooth the stresses according to a weighted average based on the filled surface (volume) of the element. If the element is close to be empty, its contribution to the average is reduced and this tends to lower the influence of the cut elements. In this case, we average the elementary stresses by considering all the element connected to the cut one. As we can see in Figure 4.24, this approach significantly reduces the peak occurring in the small elements, the highest stress appearing normally in the low left corner. Moreover, we should also mention that while being very simple, this method has the drawback of needing to build the mesh topology between elements in order to compute the averaged values. Fortunately, as the mesh is fixed, the mesh topology relations have only to be built once.

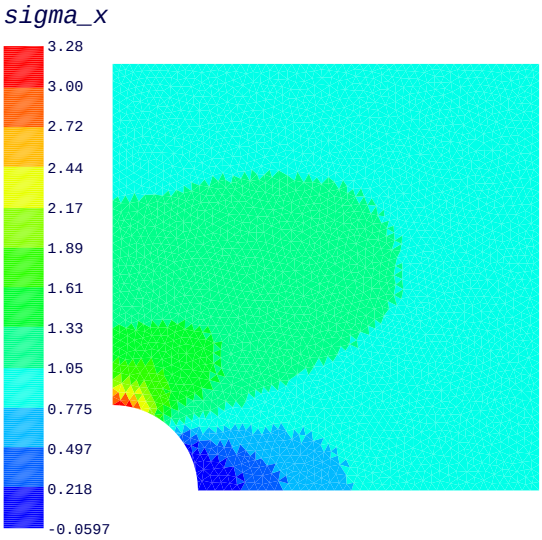


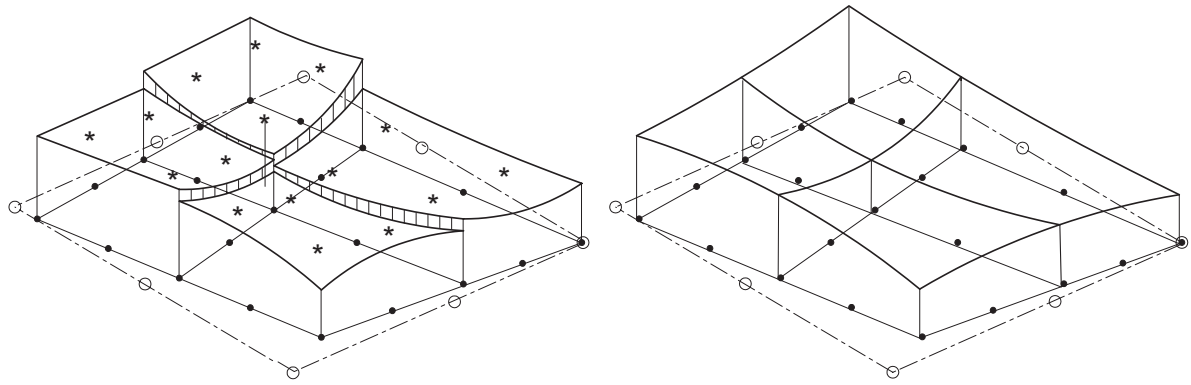

Figure 4.24: Weighted averaged stress

#### Patch recovery method applied to X-FEM

Zienkiewicz and Zhu [188] developed the idea of utilizing a local patch of elements sampled at their super-convergent points to obtain a smoothed least square fitting of nodal gradients. The super-convergent points of an element are special interior points where the gradients are normally the most accurate. They demonstrated [188] that we can generate superconvergence estimates for the stresses at the node by employing a patch (as illustrated in Figure 4.25) of elements surrounding the node. A local least square fit is generated over a set of elements constituting the patch in the following way. Let assume for the recovered stresses $\sigma_p^*$ a polynomial approximation of the form:

$$\sigma_p^* = Pa$$

where $P$ denotes a polynomial that has the same degree and the completeness that was used for the FEM approximation of the displacement and '$a$' denotes the nodal values of the stresses.



(a) Discontinuous element stresses on the patch (b) Smoothed nodal stresses

Figure 4.25: Smoothing flux values on a node based patch [7]

To compute the recovered stresses at the nodes inside a patch, the following function $F$ should be minimized :

$$F(a) = \sum_{i=1}^{n} \left(\sigma_h(\xi,\eta) - \sigma_p^*(\xi,\eta)\right)^2 = \sum_{i=1}^{n} \left(\sigma_h(\xi,\eta) - P(\xi,\eta)a\right)^2$$

where $n$ is the total number of integration points or super-convergent points if available used in the elements in the patch and $\sigma_h(\xi,\eta)$ is the numerically computed stresses. By substituting, these terms with the Finite Element approximation, we obtain on a patch:

$$F(a) \quad = \quad \sum_{e=1}^{NPE} \sum_{i=1}^{QE} \left[E^e B_j^e u^e - P_j(\xi,\eta)a\right]^2$$

where $NPE$ is the number of elements in the patch and $QE$ is the number of Gauss points used to computed $\sigma_h(\xi,\eta)$. The least square fitting over the $n$ data points enables to compute the coefficients '$a$' and the minimization of $F$ gives:

$$\sum_{i=1}^{n} P^T(\xi,\eta)P(\xi,\eta)a \quad = \quad \sum_{i=1}^{n} P^T(\xi,\eta)\sigma_h(\xi,\eta)$$

where '$a$' is a rectangular matrix of flux components at patch nodes. This system is solved for the coefficients '$a$'. The number of rows in the least square system is equal to the number of nodes defining the patch. It is therefore necessary to have a number of sampling points $n$ equal or greater than the total number of nodes on the patch (which is equal to the number of coefficients in $P$).

This method is based on the definition of a local patch. Different patch configurations can be chosen. One can define a patch by the set of adjacent elements, facing elements patch or by the elements connected to a node. These patches can be named topological if they are based on the

topology of the mesh. But one could also use a geometrical region defined by a circle or a sphere centered at a node.

Applying this recovery technique on the quarter plate with a hole problem, we can observe that the smoothing technique enables to circumvent nearly completely the overestimated stresses. The values obtained are also in agreement with the previous method.

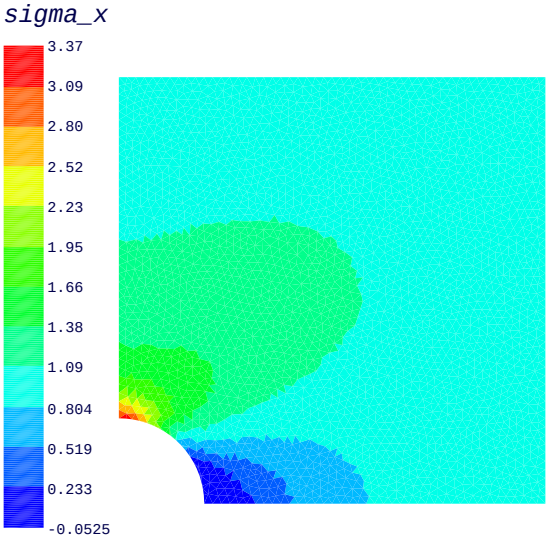
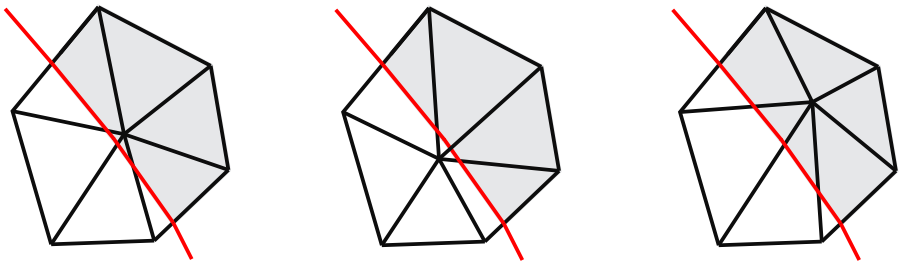



Figure 4.26: Recovered stress using patch recovery technique

### 4.3.5 Mesh modification

The last method that we have investigated is to move the mesh nodes in order to obtain a better ratio between the void and the solid part of the X-FEM elements. In this approach, we have two options: 1) move 'in material' points to enlarge the material part or 2) move 'in material' points (which are very close the interface) in the void domain.

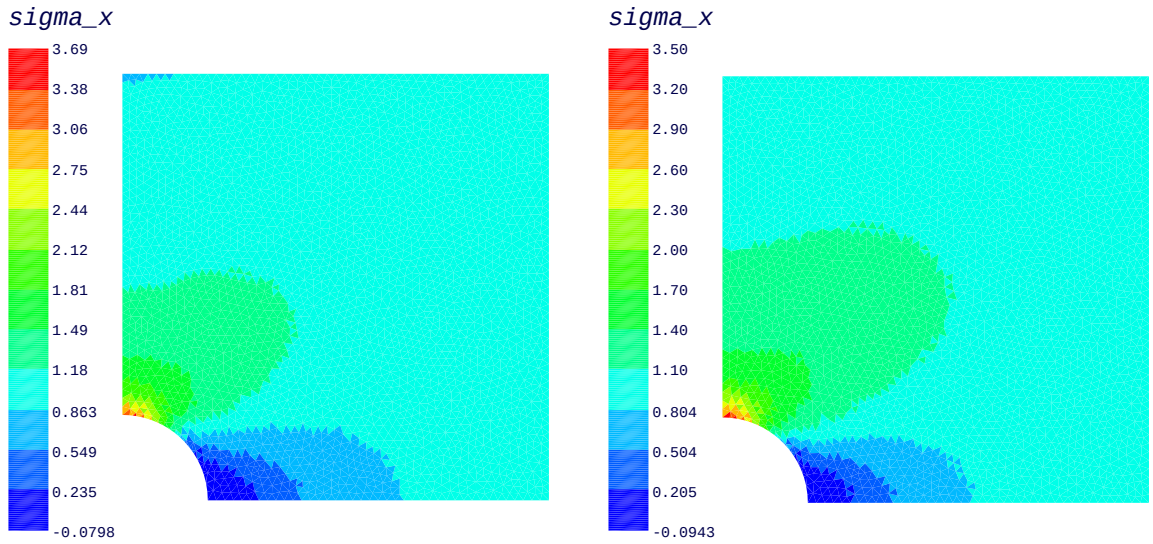


(a) Initial situation (b) Move to void (c) Move to material

Figure 4.27: Relocation of a node

The first approach consists in moving the nodes that are located inside the material domain and close to the boundary at the barycenter of the solid part of the patch composed of the

elements connected to this node. In this case, we only modify the ratio void/material of the small elements by relocating the nodes. Then, as we can obtain new elements with a higher aspect ratio as depicted on case (c) of Figure 4.27, a Laplacian smoothing is applied on the material zone while keeping the previously processed node fixed. However, as we can see in Figure 4.28, the relocation of positive nodes followed by a Laplacian creates elements that present a better void/filled ratio but with a maximum stress value bigger than with the previous methods and FEM. To further improve these results, we introduce a last step which consists in moving the negative nodes closer to the boundary and we obtain a better result as seen in Figure 4.28 (b).




(a) Relocation of positive nodes + Laplacian

(b) Relocation of positive nodes + Laplacian + relocation of negative nodes

Figure 4.28: Relocation of nodes

The second strategy considers moving the nodes out of the material domain to completely suppress the small elements. In this case, the direction of movement is given by the normal to the boundary computed from the Level Set values. As we should have a signed distance function, the intensity of the displacement is also given by the Level set. For elements presenting an edge that has 99 % of its length inside the void domain we move outside from the material zone as depicted in case (b) of Figure 4.27. We can see in Figure 4.29 that this approach provide a good result as the high peak disappeared and a stress range similar to the previous methods and the FEM solution is recovered (relative difference of 0.9 % with resepect to FEM solution). This approach seems also quite attractive because we limit the movement of the nodes according to the Level Set norm. However, as we move the node without any consideration with respect to the mesh geometry, it should be taken into consideration not to flip elements when moving nodes which can be a delicate task in 3D. When computing the sensitivity analysis with a semi-analytical scheme, this approach should be proceeded at step $x$ and $x + \delta x$ which can be time consuming. Moreover, there is no guarantee that we can cut off the erroneous stresses and do not introduce any new pathologic elements somewhere else.

Table 4.1 summarizes the different methods implemented. We can see that all approaches perform equivalently. The difference between the modified X-FEM model and FEM model is limited to

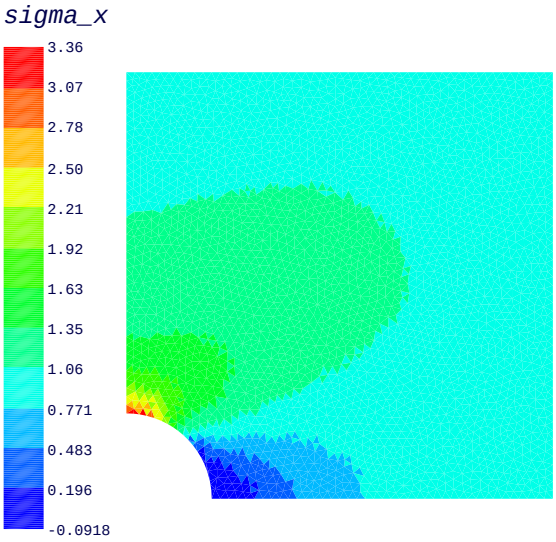



Figure 4.29: Relocating nodes to void part of the domain

2 % except when relocating positive nodes to material domain. In the chapter relative to the applications 6, the stress smoothing approach is used as a default option.

| Method | Max. stress $\sigma_x$ | Rel. Difference(%) |
|---|---|---|
| FEM | 3.33 | - |
| X-FEM | 4.03 | 21% |
| X-FEM + Node relocation to material | 3.50 | 5% |
| X-FEM + Recovered stresses | 3.27 | 1.8% |
| X-FEM + Stress smoothing | 3.28 | 1.5% |
| X-FEM + Node relocation to void | 3.36 | 0.9% |
| X-FEM + Soft material | 3.34 | 0.3% |

Table 4.1: Comparison of the maximum stress $\sigma_x$ obtained with the different approaches

## 4.4 Numerical applications

### 4.4.1 Infinite plate with a circular hole

To illustrate the accuracy and convergence properties of the X-FEM with the material-void approach we compare the numerical solution with the exact solution the problem of an infinite plate with a circular hole. The analytical solution is given in polar coordinates [169]:

$$\begin{aligned}
\sigma_{xx}(r,\theta) &= \sigma_\infty \left[1 - \frac{a^2}{r^2}\left(\frac{3}{2}\cos 2\theta + \cos 4\theta\right) + \frac{3a^4}{2r^4}\cos 4\theta\right], \\
\sigma_{yy}(r,\theta) &= -\sigma_\infty \left[\frac{a^2}{r^2}\left(\frac{1}{2}\cos 2\theta - \cos 4\theta\right) + \frac{3a^4}{2r^4}\cos 4\theta\right], \\
\tau_{xy}(r,\theta) &= -\sigma_\infty \left[\frac{a^2}{r^2}\left(\frac{1}{2}\sin 2\theta + \sin 4\theta\right) + \frac{3a^4}{2r^4}\sin 4\theta\right]
\end{aligned}$$

where $a$ is the radius of the circular hole. The displacement field is given by:

$$\begin{aligned} u_x(r,\theta) &= \frac{a}{8\mu}\left[\frac{r}{a}(\kappa+1)\cos\theta + \frac{2a}{r}((1+\kappa)\cos\theta + \cos 3\theta) - \frac{2a^3}{r^3}\cos 3\theta\right], \\ u_y(r,\theta) &= \frac{a}{8\mu}\left[\frac{r}{a}(\kappa-3)\sin\theta + \frac{2a}{r}((1-\kappa)\sin\theta + \sin 3\theta) - \frac{2a^3}{r^3}\sin 3\theta\right] \end{aligned}$$

where $\mu$ is the shear modulus and $\kappa$ the Kolosov constant which are defined as:

$$\kappa = \begin{cases} \frac{3-\nu}{1+\nu} & \text{in plane stress state} \\ 3-4\nu & \text{in plane strain state} \end{cases} \qquad \mu = \frac{E}{2(1+\upsilon)}$$

The numerical computations are realized on a square domain 1m $\times$ 1m with a traction free circular hole of radius $a = 0.3$m and a plane stress state is assumed.

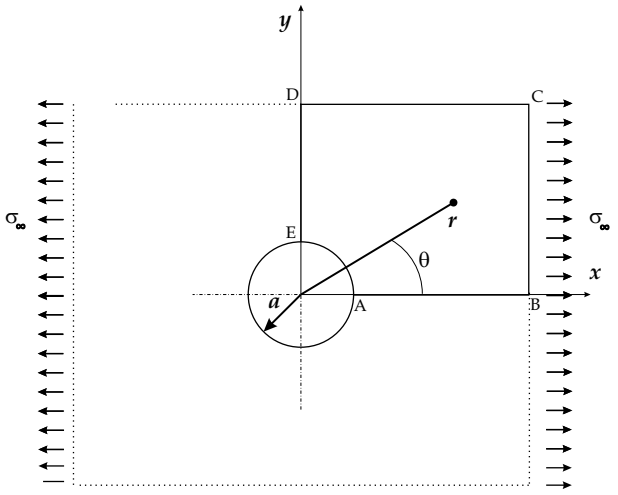



Figure 4.30: Geometry of the infinite plate with a hole

The plate is loaded on the right ($BC$) and top side ($CD$) with a loading of $\sigma_\infty$=1 N/m$^2$ and infinite plate conditions. The material considered is characterized by the following properties $E$=1 N/m$^2$and $\nu$=0.3. By symmetry, only a quarter of the plate is modeled and the essential boundary conditions are: $u_y$=0 along AB and $u_x$=0 along DE.

To compare this solution to the finite element results, the same analysis is carried with the classical FE. The error in the energy norm $\left\|\mathbf{u}-\mathbf{u}^h\right\|_{E(\Omega)}$ is defined as:

$$\begin{aligned} \left\|\mathbf{u}-\mathbf{u}^h\right\|_{E(\Omega)} &= \left(\int_\Omega (\epsilon-\epsilon^h)H(\epsilon-\epsilon^h)d\Omega\right)^{1/2} = \sum_{el}(\epsilon-\epsilon^h)H(\epsilon-\epsilon^h)\ d\Omega_{el} \\ \|\mathbf{u}\|_{E(\Omega)} &= \left(\int_\Omega \epsilon H\epsilon d\Omega\right)^{1/2} = \sum_{el}\epsilon H\epsilon\ d\Omega_{el} \end{aligned}$$

The Figure 4.32 illustrates the relative error for quadrangle finite elements while the Figure 4.31 relates to the triangular elements. The analysis is carried out with elements of order $p$=1,2,3

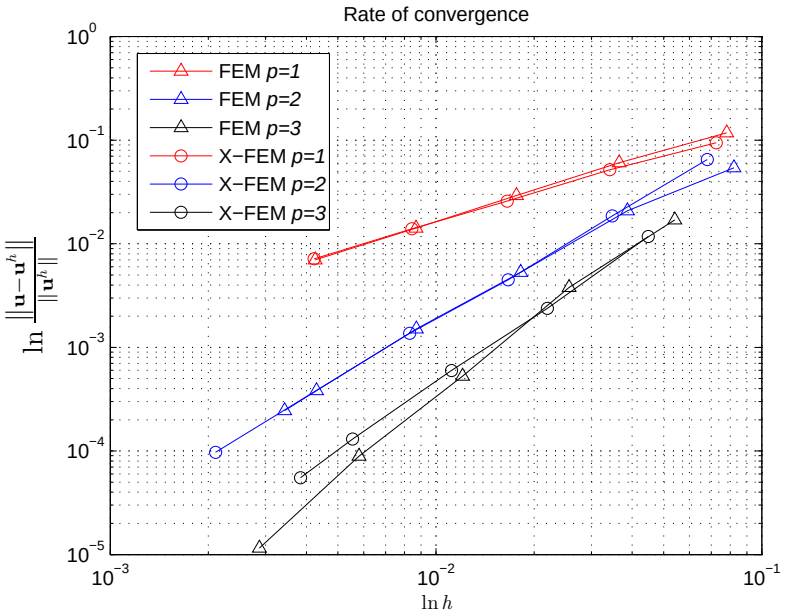



Figure 4.31: Plate with a hole: comparison of the rate of convergence for X-FEM and FEM models

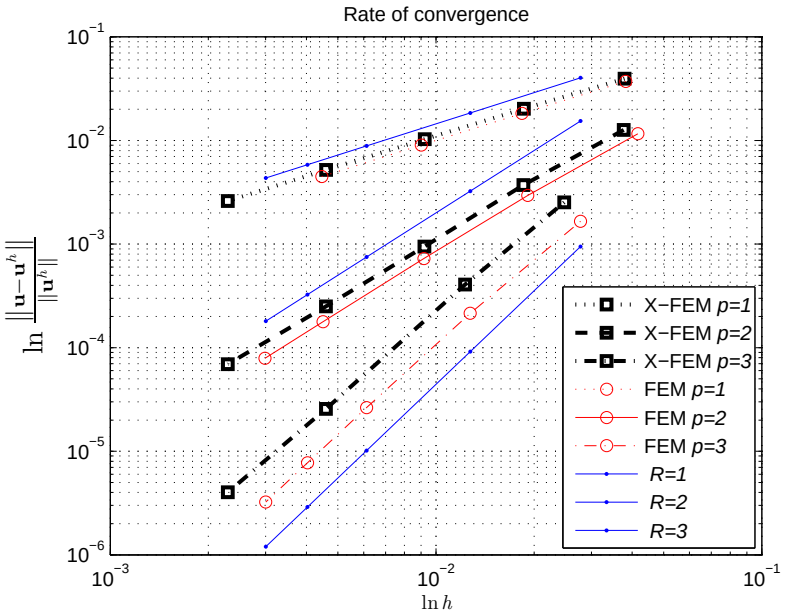



Figure 4.32: Plate with a hole: comparison of the rate of convergence for X-FEM and FEM models

using the recursive element subdivision and the integration method presented in section 4.1.7.

The agreement is very good and the theoretical rate of convergence is always very good for both quadrangular/triangular elements and approximation orders. Moreover, the rate of convergence for X-FEM and FEM are similar but the X-FEM always presents an higher error which increases with the approximation order.

### 4.4.2 Infinite plate with a circular inclusion

Let's consider the problem of an elastic circular inclusion of material properties $[\lambda_1, \mu_1]$ inside a circular body $[\lambda_2, \mu_2]$ subject to a prescribed displacement $u$ on the outer surface $\Gamma_2$.

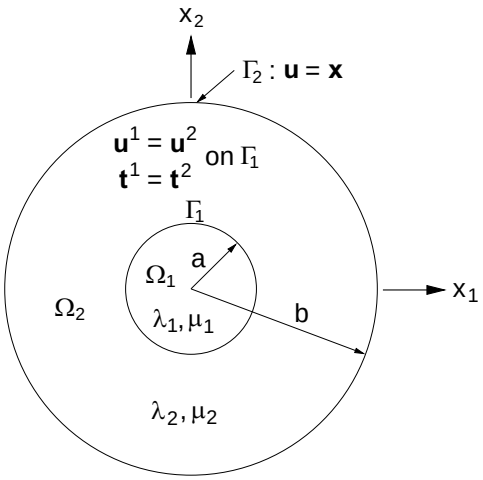



Figure 4.33: Geometry of the circular hole inclusion problem

If we assume a perfect bonding between the matrix and the inclusion i.e. $u_1(\gamma_1) = u_2(\Gamma_2)$, the analytical solution of this problem is given by:

$$\begin{aligned} u_r &= \begin{cases} \left[\left(1-\frac{b^2}{a^2}\right)\alpha+\frac{b^2}{a^2}\right] r, & 0 \le r \le a \\ \left(r-\frac{b^2}{r}\right)\alpha+\frac{b^2}{r}, & a \le r \le b \end{cases} \qquad u_\theta = 0 \\ \alpha &= \frac{(\lambda_1+\mu_1+\mu_2)\, b^2}{(\lambda_2+\mu_2)\, a^2+(\lambda_1+\mu_1)\,(b^2-a^2)+\mu_2 b^2} \\ \lambda &= \frac{E\nu}{(1-v)\,(1-2v)} \end{aligned}$$

where $\lambda$ and $\mu$ are the Lamé constants. The strains and stresses are:

$$\varepsilon_{rr}(r) = \begin{cases} \left(1-\frac{b^2}{a^2}\right)\alpha+\frac{b^2}{a^2}, & 0 \le r \le a \\ \left(1-\frac{b^2}{r^2}\right)\alpha-\frac{b^2}{r^2}, & a \le r \le b \end{cases} \quad \text{and} \quad \varepsilon_{\theta\theta}(r) = \begin{cases} \left(1-\frac{b^2}{a^2}\right)\alpha+\frac{b^2}{a^2}, & 0 \le r \le a \\ \left(1-\frac{b^2}{r^2}\right)\alpha+\frac{b^2}{r^2}, & a \le r \le b \end{cases}$$

$$\sigma_{rr}(r) = \begin{cases} (2\mu+\lambda)\left[\left(1-\frac{b^2}{a^2}\right)\alpha+\frac{b^2}{a^2}\right]+\lambda\left[\left(1-\frac{b^2}{a^2}\right)\alpha+\frac{b^2}{a^2}\right], 0 \le r \le a, \\ (2\mu+\lambda)\left[\left(1-\frac{b^2}{r^2}\right)\alpha-\frac{b^2}{r^2}\right]+\lambda\left[\left(1-\frac{b^2}{r^2}\right)\alpha+\frac{b^2}{r^2}\right], a \le r \le b, \end{cases}$$

$$\sigma_{\theta\theta}(r) = \begin{cases} (2\mu+\lambda)\left[\left(1-\frac{b^2}{a^2}\right)\alpha+\frac{b^2}{a^2}\right]+\lambda\left[\left(1-\frac{b^2}{a^2}\right)\alpha+\frac{b^2}{a^2}\right], 0 \leq r \leq a, \\ (2\mu+\lambda)\left[\left(1-\frac{b^2}{r^2}\right)\alpha+\frac{b^2}{r^2}\right]+\lambda\left[\left(1+\frac{b^2}{r^2}\right)\alpha-\frac{b^2}{r^2}\right], a \leq r \leq b, \end{cases}$$

For symmetry reasons, the tangential displacement and the shear components of the strain and stress tensors are equal to zero.

In the numerical computation, the domain studied is a 1m $\times$ 1m square with a circular inclusion with a radius $a$=0.4 m and proper symmetry boundary conditions are applied. On the boundary of the plate, we apply the exact stresses to simulate a circular domain. This study is realized solely with first degree triangular finite elements. As previously presented, we use here the ridge enrichment function $g(x)$ proposed by Moës *et al.* [113]:

$$g(x) = \sum_i |\psi_i| N_i(x) - \left|\sum_i \psi_i N_i(x)\right|$$

The results of the convergence of the energy error norm is shown in Fig. 4.34. The material in the zone 1 has a Young modulus $E_1$=1 N/m$^2$ and Poisson ratio of 0.25 while the second zone is characterized by $E_2$=10 N/m$^2$ and $\nu_1$=0.3. The convergence analysis is carried out with three models: 1/FEM conforming to the material interface, 2/ FEM non conforming to the interface and 3/ X-FEM.

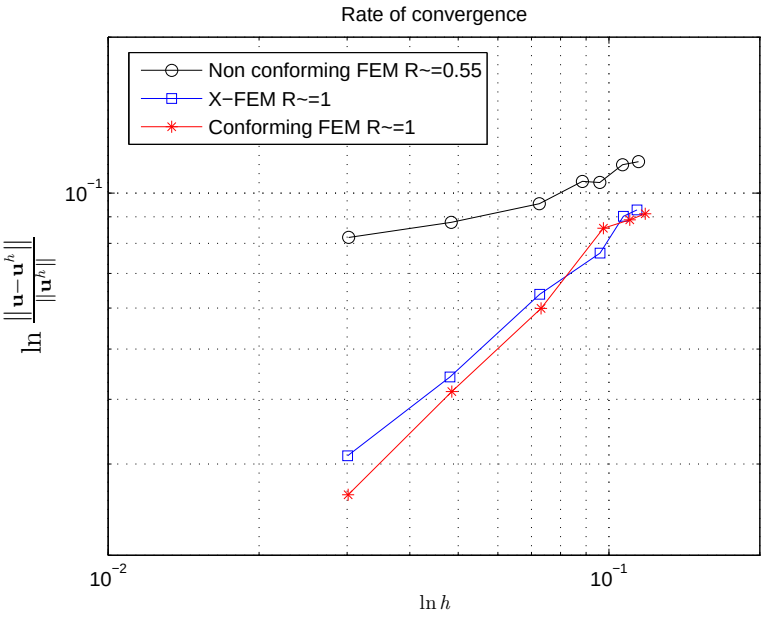



Figure 4.34: Convergence of the error in energy norm

Here also, we can see in Figure 4.34 that the X-FEM (squared markers) performs well compared to the conforming FEM (stared markers) and that the rate of convergence is close to the theoretical rate of 1. With a very coarse mesh, the solution is even better with X-FEM than with conforming FEM. However, when the mesh size tends to decrease, the conforming FEM exhibits a slightly better convergence rate and is more accurate in terms of error. As expected, the non conforming FEM performs very poorly in terms of error and convergence as it can be observed on the curve with circle markers.

### 4.4.3 Natural vibration and eigenvalues

The procedure described previously for the integration of the stiffness matrix has been extended to the integration of the mass matrix in order to evaluate the eigenfrequencies and eigenmodes with X-FEM. The Table 4.2 compares the eigenfrequencies computed with a triangular first degree FEM model and a relative triangular X-FEM model. The same results are also given for quadrangle elements (see Fig. 4.35 (b) and Table 4.3).

| Mode | FEM | X-FEM | Rel. diff. % |
|---|---|---|---|
| 1 | 0,60553 | 0,6045 | 0,1572 |
| 2 | 0,62198 | 0,6217 | 0,0403 |
| 3 | 1,2885 | 1,2879 | 0,0442 |
| 4 | 1,33395 | 1,3334 | 0,0404 |
| 5 | 1,33433 | 1,3338 | 0,0329 |
| 6 | 1,47887 | 1,4776 | 0,0845 |
| 7 | 1,47944 | 1,4793 | 0,0094 |
| 8 | 1,59345 | 1,5936 | 0,0094 |
| 9 | 2,14034 | 2,1401 | 0,0079 |
| 10 | 2,32701 | 2,3249 | 0,0876 |

Table 4.2: Eigenfrequencies of FEM and X-FEM with triangular mesh model.

| Mode | FEM | X-FEM | Rel. diff. (%) |
|---|---|---|---|
| 1 | 0,59566 | 0,59430 | 0,22848 |
| 2 | 0,60586 | 0,60565 | 0,03499 |
| 3 | 1,28414 | 1,2846 | 0,03582 |
| 4 | 1,31874 | 1,31645 | 0,17365 |
| 5 | 1,31912 | 1,31645 | 0,20240 |
| 6 | 1,46138 | 1,45961 | 0,12111 |
| 7 | 1,46206 | 1,45961 | 0,16757 |
| 8 | 1,58777 | 1,58644 | 0,08376 |
| 9 | 2,09945 | 2,08551 | 0,66398 |
| 10 | 2,28669 | 2,27859 | 0,35422 |

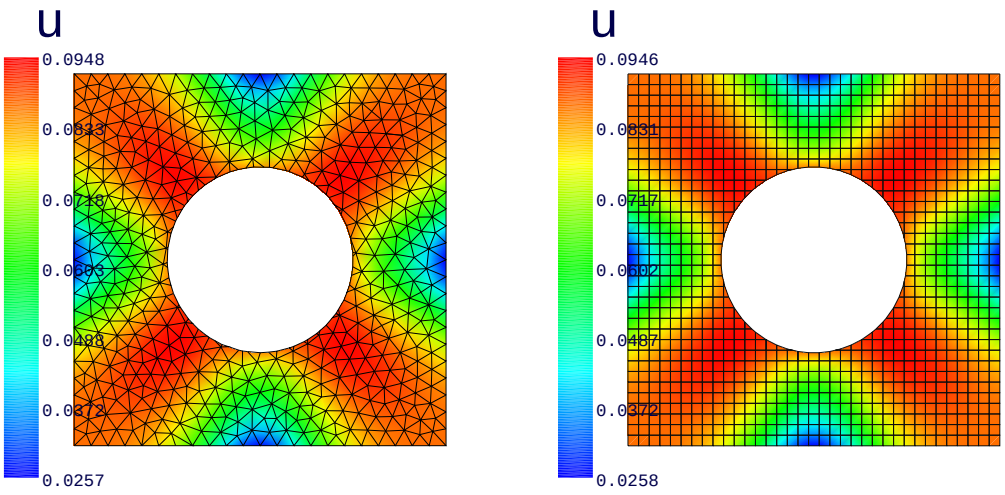
Table 4.3: Eigenfrequencies of FEM and X-FEM quadrangle mesh model.

We can remark that the agreement between the classical method and the extended FE is excellent as the relative difference is always smaller than 0.2%. We can draw the same conclusions for quadrangle elements (Table 4.3). However, we can notice that the difference is slightly bigger (0.66%) than for the triangular mesh. This difference is probably due to the fact that the FEM and the X-FEM quadrangle meshes are differents. The X-FEM mesh is shown in Figure 4.35 (b) while the FEM mesh is illustrated in Fig. 4.36 (a).

The relative difference between FEM and X-FEM eigenmode is presented in the Figure 4.36.
A comparison of the convergence of the first natural frequency for four different models is present in Figure 4.37. As noticed, the results from the two triangular meshes are always very close whereas the structured mesh models present a larger difference. Finally, we can also notice that the X-FEM obtains, for each level of refinement, a natural frequency which is lower than the FEM one indicating that X-FEM provides slightly softer and so better solutions than FEM.

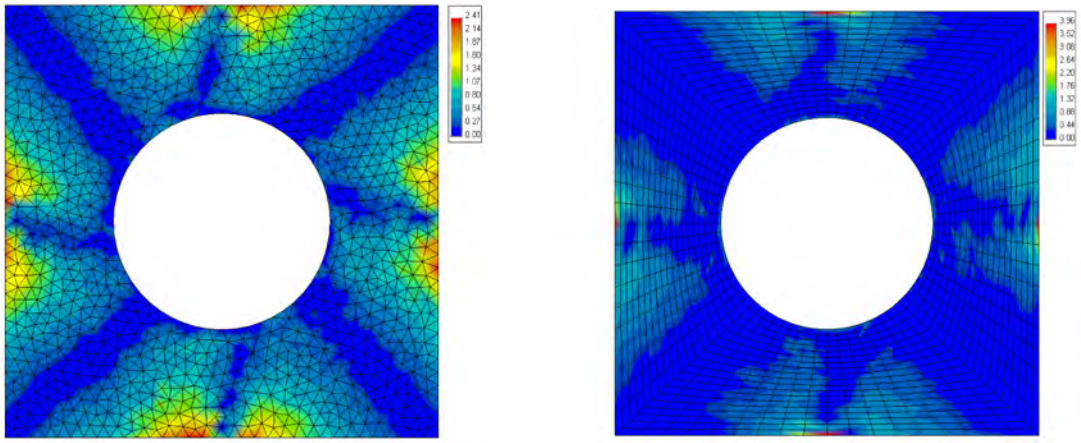
## 4.5 Embedded boundary conditions

The extended finite element method generalizes the finite element method to handle domains whose geometrical boundaries are not necessarily covered by a conforming mesh. Therefore, one can choose either to mesh the whole domain conforming to all boundaries or just a part of the boundaries. Generally, the geometry of the structure being given, the mesh is thus conforming to the geometry and the discontinuities are embedded inside the domain. However, in certain situations, it can be useful to prescribe a boundary condition or an interface condition between two media. With X-FEM, this means that the boundary conditions have to be applied inside the mesh along the interface or the discontinuity. For instance one can apply geometry-depending loads (e.g. pressure) on the boundary and to study the influence of this boundary shape or also prescribe a voltage for a model described with an implicit Level Set representation with non




(a) Eigenmode with triangular X-FEM elements (b) Eigenmode with quadrangular X-FEM elements

Figure 4.35: X-FEM eigenmodes



(a) Triangular elements (b) Quadrangular elements

Figure 4.36: Comparison between X-FEM and FEM eigenmodes

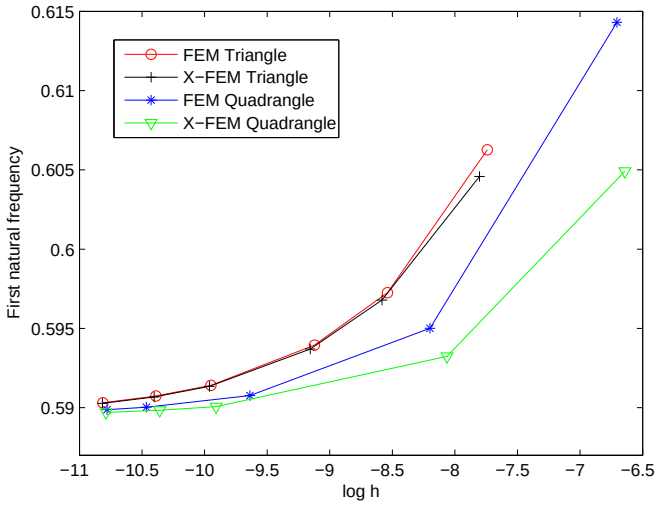



Figure 4.37: Comparison of the first natural frequency obtained with X-FEM and FEM

conforming mesh using X-FEM.

On the one hand, imposing Neumann or traction-free boundary condition presents little difficulties as it just requires a modification of the integration procedure with respect to the finite element approximation. On the other hand, the imposition of Dirichlet boundary conditions has been a research issue since the very beginning of the PUM methods.

### 4.5.1 Neumann boundary condition with X-FEM

In linear static problems, the expression of the external loads are given (if we omit volumic loads) by:

$$\mathbf{f} = \int_{\Gamma_t} \overline{\mathbf{t}} v d\Gamma_t$$

where $v$ are test functions, $\Gamma_t$ the portion of the boundary that support the applied load and $\overline{\mathbf{t}}$ the lineic or surfacic load vector. In Galerkin method, this leads to the following formulation of the generalized loads when using the finite element method, with $\Gamma_t^h$ the FE discretization of $\Gamma_t$:

$$\mathbf{f}_i = \int_{\Gamma_t^h} \overline{\mathbf{t}} N_i(\mathbf{x}) d\Gamma_t^h \tag{4.8}$$

When dealing with a classical finite element method, the boundary $\Gamma_t^h$ is always conforming to the finite elements mesh. Hence, the integral (4.8) is always evaluated along the border or the face of the elements. As a consequence, because of the Kronecker-$\delta$ properties of the finite element approximations, (4.8) has to be evaluated only for the node belonging to the edge (or the face in 3D) on which the boundary conditions are applied. Practically, to apply a Neumann boundary conditions in FE codes, one develops specific elements such as pressure elements. These elements are placed all over the $\Gamma_t^h$ boundary with associated shape function that corresponds to the trace of the shape functions of the finite element domain. The integration points are then defined on these boundary elements.

In the extended finite element method, the boundary $\Gamma_t^h$ is not explicitly represented by a set of element borders but can lie inside the elements. Therefore, (4.8) still holds but, the numerical integration procedure should be modified as we have to integrate the shape functions inside the element along the interface and not only on the borders of this element. As a consequence, all the nodes of the X-FEM element are loaded whereas only the nodes belonging to the interface $\Gamma_t^h$ has a defined loads with FEM.

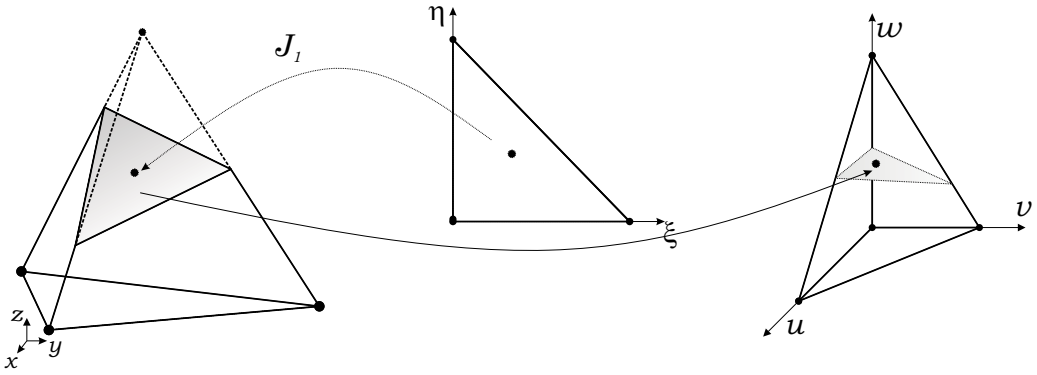



Figure 4.38: Mapping triangle Gauss point into tetrahedron

Following the method used to integrate the stiffness matrix explained in chapter 4.1, the integration of the loads needs two different mappings. A first mapping is needed to determine the position of the Gauss points of the interface in the physical space of the tetrahedron. These points are taken from a classical Gauss integration rule defined on the interface. For instance, if the interface is triangular, as shown in Figure 4.38, one should use an integration rule defined for this geometry and adapted to the degree of the shape functions to integrate (i.e. tetrahedron shape functions). Given these points mapped into the physical space, the relative Jacobian and the integration weights, a second mapping is necessary to evaluate the shape function $N_i(x)$ of the parent element (intersected tetrahedron in Figure 4.38). Hence, the expression of the loads is computed as:

$$\mathbf{f}_i = \sum_j^{n^{gp}} \overline{\mathbf{t}} N_i(u^j, v^j, w^j) \omega^j \, |J_1| \tag{4.9}$$

where $J_1$ is Jacobian of the transformation from the reference triangle to the tetrahedron physical space, $(u^j, v^j, w^j)$ the Gauss points in $(u, v, w)$ and $\omega^j$ are the weights defined in the Gauss quadrature rule.

When the boundary condition is applied on all the interfaces, the procedure is numerically simple as we only have to loop over all the iso-zero Level Set interface. However, when the model consists of more then one Level Set, it is necessary to sort the different interfaces according to a Level Set identifier. It is also more convenient to store the iso-zero Level Set as a set of special finite element that keeps a link to their parents.

**Application: thick-walled cylinder under uniform boundary pressure**

We consider a hollow cylinder of internal and external radius $a$ and $b$, respectively. An internal uniform pressure $p_i$ is applied on the internal hole ($r$=$a$) while the external surface ($r$=$b$) is free

($p_e$=0), see Fig. 4.39. Due to revolution symmetry, the solution is independent of the angle $\theta$ and the stress field is a function of the radius $r$. In the case of plane strain state, the exact solution for the stress field is given in polar coordinates ($r$,$\theta$) by [145]:

$$\begin{aligned}
\sigma_r(r) &= \frac{a^2p_i - b^2p_e}{b^2 - a^2} - \frac{p_i - p_e}{\frac{r^2}{a^2} - \frac{r^2}{b^2}} \\
\sigma_z(r) &= \nu(\sigma_r + \sigma_\theta) = 2\nu\frac{a^2p_i - b^2p_e}{b^2 - a^2} \\
\sigma_\theta(r) &= \frac{a^2p_i - b^2p_e}{b^2 - a^2} + \frac{p_i - p_e}{\frac{r^2}{a^2} - \frac{r^2}{b^2}} \\
\sigma_{r\theta}(r) &= 0
\end{aligned}$$

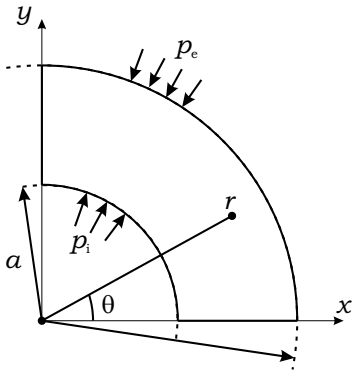



Figure 4.39: Thick-walled cylinder problem

and the displacement are:

$$\begin{aligned}
u_r(r) &= \frac{1+\nu}{E}\left[-\frac{a^2b^2\,(p_e - p_i)}{b^2 - a^2}\frac{1}{r} + (1-2\nu)r\frac{a^2p_i - b^2p_e}{b^2 - a^2}\right] \\
u_\theta, u_z &= 0
\end{aligned}$$

Due to axisymmetry condition, only a quarter of the cylinder is modeled by imposing suitable boundary conditions on the $x$ and $y$ axis. In the model, the parameters are the following: $a$=0.4 m, $b$=1 m $p_i$=1 N/m$^2$ $p_e$=0, $E$=10e$^5$ N/m$^2$, $\nu$=0.3. The convergence analysis is realized using five different mesh refinement with first degree quadrangular and triangular elements. For the transfinite meshes, the domain is discretized using $L_h \times L_h$ elements with $L_h$=[10, 20, 40, 80, 160] while a decreasing $h$ size for triangular elements is used. We compare both the accuracy of the FEM and X-FEM solution with respect to the analytical solution in terms of the error in the energy norm.

The convergence curves for the different models are presented in Fig. 4.40 (where the triangle markers and square markers refer to triangular and quadrangular elements respectively). The continuous blue line represent the theoretical optimal asymptotic rate of convergence. One can see that the rate of convergence of both FEM and X-FEM is very close to the optimal rate of convergence of 1. Moreover, there is no significant difference between the error obtained with X-FEM and FEM.

### 4.5.2 Dirichlet boundary condition with X-FEM

With FEM, the Dirichlet boundary conditions are directly imposed on the boundary because of Kronecker-$\delta$ property of the shape functions. With X-FEM, two different configurations can occur: the boundary conditions can be prescribed on the model boundary (along the element boundaries) or inside the mesh. In the first case, if the enrichment function is not shifted, the Kronecker-$\delta$ is not fulfilled ($u(x_i) \neq u_i$) and prescribing the boundary conditions on the element boundary is difficult. This situation is similar to what occurs in meshfree approximations where

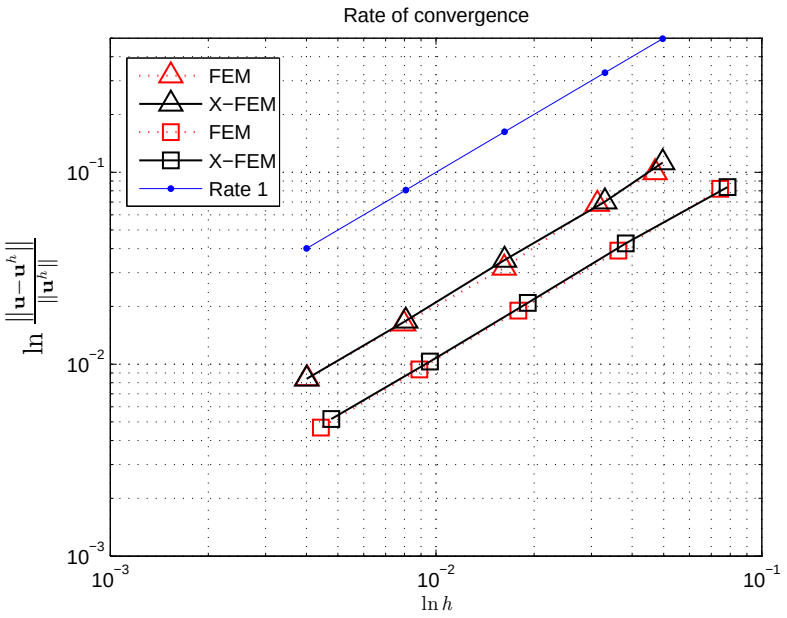
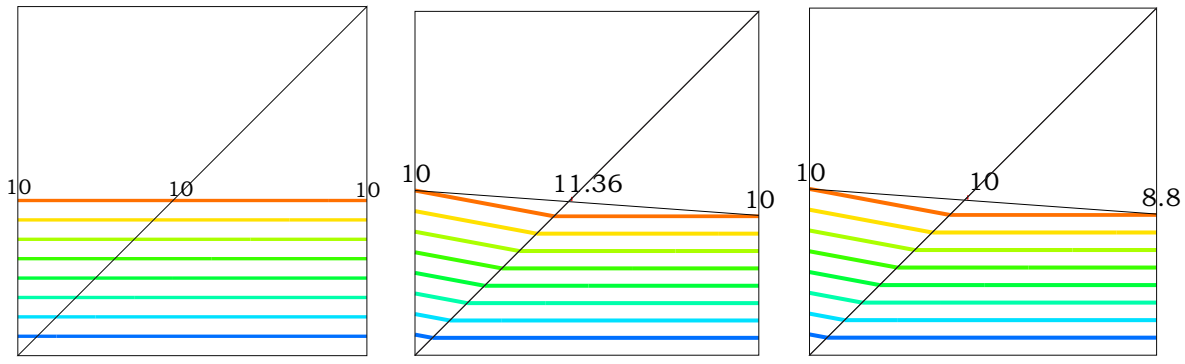



Figure 4.40: Rate of convergence for the thick-walled hollow cylinder problem

several shape functions can overlap. It is then useful to enforce the conditions weakly. The second configuration arises when the boundary condition is enforced inside the mesh. In this case it is not possible, in the most general way, to prescribe boundary conditions by imposing the nodal values on the phantom nodes. For instance if we consider a rectangular material domain in two linear triangle elements as shown in Figure 4.41, imposing zero volt on the lower edge and a voltage $\overline{\phi}$= 10 V along the interface $\gamma$ is not possible in every configurations. One can observe in Figure 4.41 that the electric potential is accurately prescribed at the interface only when this one is parallel to the bottom electrode. However when the interface is not parallel to an element edge, a constant potential at the interface can not be obtained.




(a) Horizontal interface (b) Nodal value imposition (c) Penalty method

Figure 4.41: Imposed potential when the electrodes are parallel and oblique.

The issues are similar to those encountered with meshfree methods. Currently, the impositions of the essential boundary conditions have been investigated by resorting to one of the following approaches [70]:

- Penalty method;
- Lagrange multipliers method.

In the *penalty method*, a quadratic penalty term is introduced in the original energy $W$ of the element:

$$W = W_e + \frac{1}{2} p \int_{\Gamma_d} \Gamma^T \Gamma \, d\Gamma_d$$

where $p$ is the penalty factor, $\Gamma = \phi - \bar{\phi}$, $\Gamma_d$ the domain of imposition and $W_e$ is the 'normal' element energy. In the penalty method, the imposition of the constraint does not add any unknowns but consists in adding a term to the stiffness matrix and external loads to the element. The drawback of this approach is that determining the optimal value of the penalty parameter $p$ is not straightforward is problem dependent. Moreover, high values of $p$ are necessary to prescribe the boundary conditions, which generally leads to a ill-conditioned systems. In X-FEM, this approach has been used by Chessa *et al.* [45] to treat solidification problems.

In [112], Moës *et al.* proposed to use a Lagrange multipliers method. Additional unknowns associated to the Lagrange multipliers are present and a mixed formulation is obtained. In this approach, it is necessary to define the space of the dual field relative to the Lagrange multipliers. This choice is not unique and it is shown in [112] that the selection of the Lagrange multipliers space is not trivial. Indeed, [88] and [112] have shown that adding multipliers at each intersection point obtained by cutting the interface with element edge may lead to oscillations and a reduction of the convergence rate. To circumvent this problem two approaches have been proposed. The approach of Moës *et al.* [112] is to reduce the space of Lagrange multipliers by selecting only a subset of the intersection points. They proposed in [112] an algorithm to select which points to consider. This one has proven to achieve optimal rates of convergence and a theoretical analysis is given in [15]. The second strategy considers the introduction of a stabilization of the Lagrange multipliers [121].

In the present work, most applications have been realized without the need to prescribe embedded Dirichlet conditions and we have limited ourselves to the implementation of the Penalty method and two stabilized Lagrange multiplier methods. The implemented approach has been proposed by Bricteux and Remacle [39] and is based on the work of Barbosa and Hugues [13]. The first method consists in adding a Laplace stabilization term $\frac{1}{2}\int p\,|\nabla\lambda|^2$ for the Lagrange multiplier whereas the second adds a least square stabilization term as proposed in [82].

Furthermore, we have developed with Dr. Véronique Rochus [144] a dedicated triangular element for strong imposition of boundary conditions. The central idea of this approach is to create a bi-material X-FEM element with specific shape functions that provide the flexibility to prescribe strongly the boundary conditions. Initially developed as an element able to model iso-potential interface conditions in bi-material electromechanical problem, the approximation field can also be used to prescribe the essential boundary conditions along material-void interfaces. In this case, as no additional shape function are present, the element can be seen as an intrinsic X-FEM. Currently, this approach has been applied to linear triangle element and the extension to 3D and other types of element is not straightforward. The details of this element is presented in

the section 4.5.3 for the bi-material approach and in section 4.5.4 for material-void iso-potential boundary conditions.

### 4.5.3 Enriched element for iso-potential bi-material boundary conditions

The underlying idea of this novel approach can be understood by observing that if the extended element was build out of two finite elements (one for the conductor and one for the non-conducting part), it would be straightforward to prescribe the boundary conditions on the interface. So, let's create an element with enriched shape function based on quadrangular shape functions for the quadrangular part and on triangular shape functions for the triangular part. To this end, we introduce two successive changes of variables: a first one between $(x, y)$ physical space and $(\xi, \eta)$, and a second one between $(\xi, \eta)$ and $(s, t)$, as sketched in Figure 4.42.

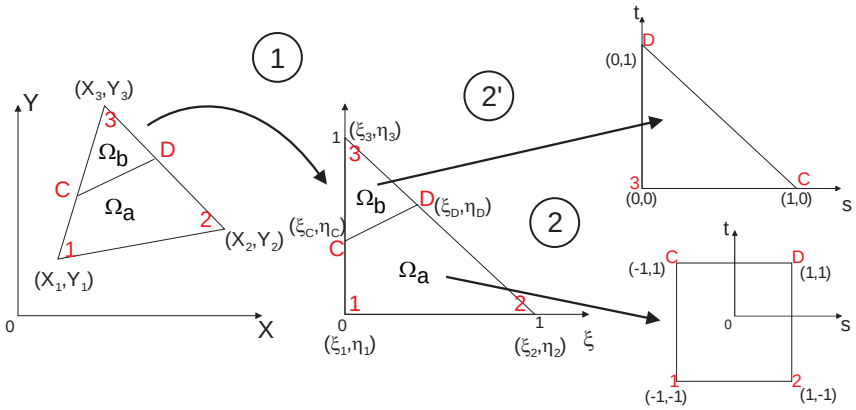



Figure 4.42: Successive transformations for an extended triangular element.

#### First change of variables

The first change of variables is the standard mapping from the physical space $(x, y)$ and $(\xi, \eta)$. It is identical for both domains $\Omega_a$ and $\Omega_b$ (see Figure 4.42).

#### Second Transformation - Quadrangular Part $\Omega_a$

The usual shape functions for a quadrangle are:

$$\begin{cases} L_1 = (1-s)(1-t)/4 \\ L_2 = (1+s)(1-t)/4 \\ L_3 = (1+s)(1+t)/4 \\ L_4 = (1-s)(1+t)/4 \end{cases}$$

The mapping between the reference space $(s, t)$ and the intermediate space $(\xi, \eta)$ is given by the following isoparametric transformation:

$$\begin{cases} \xi(s,t) = \xi_1 L_1 + \xi_2 L_2 + \xi_D L_3 + \xi_C L_4 \\ \eta(s,t) = \eta_1 L_1 + \eta_2 L_2 + \eta_D L_3 + \eta_C L_4 \end{cases}$$

The last two shape functions are used to enhance the solution field since the shape functions $L_1$ and $L_2$, associated to the nodes 1 and 2, are unnecessary as the element already bears the

shape functions $N_1$ and $N_2$ for these nodes. Therefore, we define the total discretization of the extended field in $\Omega_a$ as:

$$\begin{array}{rl} \phi_a = & N_1\left(\xi(s,t),\eta(s,t)\right)\Phi_1 + N_2\left(\xi(s,t),\eta(s,t)\right)\Phi_2 \\ & +N_3\left(\xi(s,t),\eta(s,t)\right)\Phi_3 + L_4(s,t)\Phi_1^* + L_3(s,t)\Phi_2^* \end{array} \tag{4.10}$$

where $\Phi_1^*$ and $\Phi_2^*$ are enriched DOFs acting on the interface.

**Second Transformation - Triangular Part $\Omega_b$**

The shape functions for the triangular domain are the same as the initial change of variables:

$$\begin{cases} G_1 = (1-s-t) \\ G_2 = s \\ G_3 = t \end{cases}$$

and the isoparametric transformation becomes:

$$\begin{cases} \xi(s,t) = \xi_3 G_1 + \xi_C G_2 + \xi_D G_3 \\ \eta(s,t) = \eta_3 G_1 + \eta_C G_2 + \eta_D G_3 \end{cases}$$

As for the quadrangular part, only the shape functions that are not associated to the basic nodes (here $G_2$ and $G_3$) define the enrichment field. The total enhanced shape functions are thus, for domain $\Omega_b$:

$$\begin{array}{rl} \phi_b = & N_1\left(\xi(s,t),\eta(s,t)\right)\Phi_1 + N_2\left(\xi(s,t),\eta(s,t)\right)\Phi_2 \\ & +N_3\left(\xi(s,t),\eta(s,t)\right)\Phi_3 + G_2(s,t)\Phi_1^* + G_3(s,t)\Phi_2^* \end{array} \tag{4.11}$$

Obviously since the added degrees of freedom $\Phi_1^*$ and $\Phi_2^*$ used in (4.10) and (4.11) are identical, along the interface $C, D$, the continuity of the field is guaranteed while the derivative can be discontinuous due to the enrichment field.

**Resulting discretization**

Finally both discretizations can be summarized in a compound triangle element in the following way:

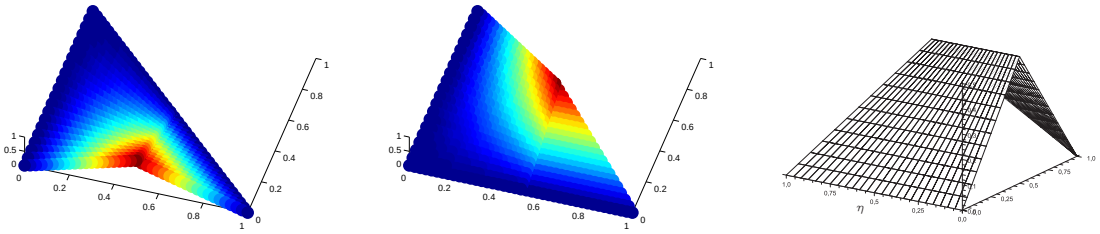
$$\left\{ \ \phi = N_1\Phi_1 + N_2\Phi_2 + N_3\Phi_3 + M_1\Phi_1^* + M_2\Phi_2^* \right. \tag{4.12}$$

$$\begin{cases} N_1 = (1-\xi-\eta) \\ N_2 = \xi \\ N_3 = \eta \end{cases} \quad \begin{cases} M_1 = s \\ M_2 = t \end{cases} \text{ in } \Omega_b \quad \begin{cases} M_1 = (1-s)(1+t)/4 \\ M_2 = (1+s)(1+t)/4 \end{cases} \text{ in } \Omega_a \tag{4.13}$$

which is represented in Figure 4.43 (a) and (b).

It is easy to verify that a hat function $\Theta$ (illustrated in Figure 4.43 (c)) is retrieved on domain $\Omega_a$ and $\Omega_b$ when $\Phi_1^* = \Phi_2^*$:

$$\Theta = \begin{cases} L_3 + L_4 & \text{in } \Omega_a \\ G_2 + G_3 & \text{in } \Omega_b \end{cases}$$



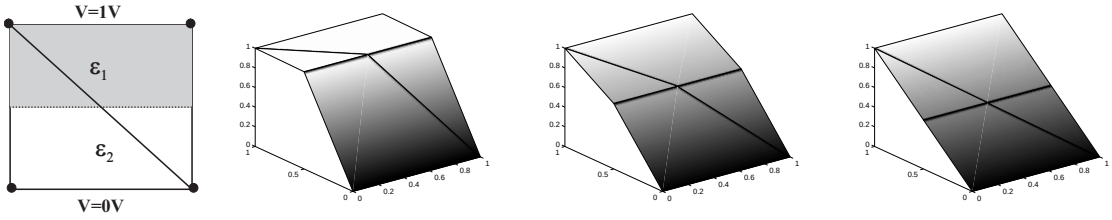
(a) $M_1$ shape function (b) $M_2$ shape function (c) Enrichment of domain $a$.

Figure 4.43: New enriched shape functions.

### Numerical Validation of the Extended Finite Element

The novel extended electrostatic finite element is validated first on a simple test case. A unit square domain is subject to a 1 Volt difference of voltage between the lower edge (ground) and the top domain (conducting domain) as shown in Figure 4.44. The domain is modeled with 2 triangular extended finite elements equivalently to what is done in Figure 4.41.

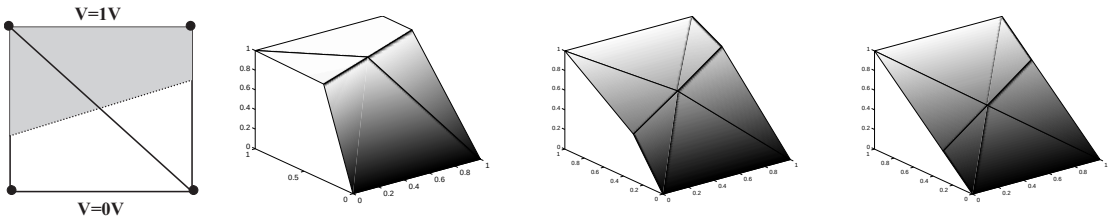
First the interface between conductor and the vacuum is taken parallel to the ground electrode. The computed electric potential is plotted in Figure 4.44: it is observed that the potential is constant on the conductor and decreases linearly between the electrodes as expected. The upper part of the problem has a permittivity of $\varepsilon_1$ and the lower one $\varepsilon_2$. Three different situations are simulated and compared to the analytical solutions. The first one is when $\varepsilon_1 \gg \varepsilon_2$, this situation appears when part 1 is a conductive material. The electric potential is nearly constant over this area as shown in Figure 4.44. The second simulation is in the case where $\varepsilon_1 = 2\varepsilon_2$. The potential at the middle point corresponds to the analytical value. Finally the last case show that a constant electrostatic field may be simulated if $\varepsilon_1 = \varepsilon_2$.




(a) (b) For $\varepsilon_1 \gg \varepsilon_2$ (c) For $\varepsilon_1 = 2\varepsilon_2$ (d) For $\varepsilon_1 = 2\varepsilon_2$

Figure 4.44: Potential for different values of $\varepsilon$.

The electric potential is also recovered if the interface is not parallel to the extremities as illustrated in Figures 4.45 for the same material properties. As shown in Figure 4.45 the potential iso-values are again piecewise linear and clearly the new shape functions of the eXtended Finite Elements can properly handle the computation of the electric potential even if the interface is oblique.




(a) (b) For $\varepsilon_1 \gg \varepsilon_2$ (c) For $\varepsilon_1 = 2\varepsilon_2$ (d) For $\varepsilon_1 = 2\varepsilon_2$

Figure 4.45: Potential for different values of $\varepsilon$.

### 4.5.4 Reduction of the approximation field to prescribe boundary conditions for conductive material

There are different strategies to model the interface between a conductive material with a bi-material X-FEM element. The first method consists in using a large value for the permittivity of the conductive material. The electric voltage is constant on the conductive domain and behaves like a classical FEM element on the dielectric part. In that case, each element that is cut by the interface has five degrees of freedom as indicated in expression (4.12). The second approach is to prescribe the derivatives of the potential on the conductive part to be equal to zero. From the expression (4.12), the derivative may be computed as follows:

$$\text{if } \Omega_a \text{ is conductive} \begin{cases} \frac{\partial \phi_a}{\partial s} = 0 \\ \frac{\partial \phi_a}{\partial t} = 0 \end{cases} ; \quad \text{if } \Omega_b \text{ is conductive} \begin{cases} \frac{\partial \phi_b}{\partial s} = 0 \\ \frac{\partial \phi_b}{\partial t} = 0 \end{cases}$$

From these expressions we obtain two relations between the five unknowns.

$$\text{if } \Omega_a \text{ is conductive} \begin{cases} \Phi_C^* = (\Phi_1 - \Phi_3)\eta_C \\ \Phi_D^* = (\Phi_1 - \Phi_3)\eta_D \end{cases} \quad \text{if } \Omega_b \text{ is conductive} \begin{cases} \Phi_C^* = (\Phi_3 - \Phi_1)(1 - \eta_C) \\ \Phi_D^* = (\Phi_3 - \Phi_2)(1 - \eta_D) \end{cases}$$

Using these relations we can reduce the number of degrees of freedom to the nodal unknowns :

$$\phi = N_1^* \Phi_1 + N_2^* \Phi_2 + N_3^* \Phi_3$$

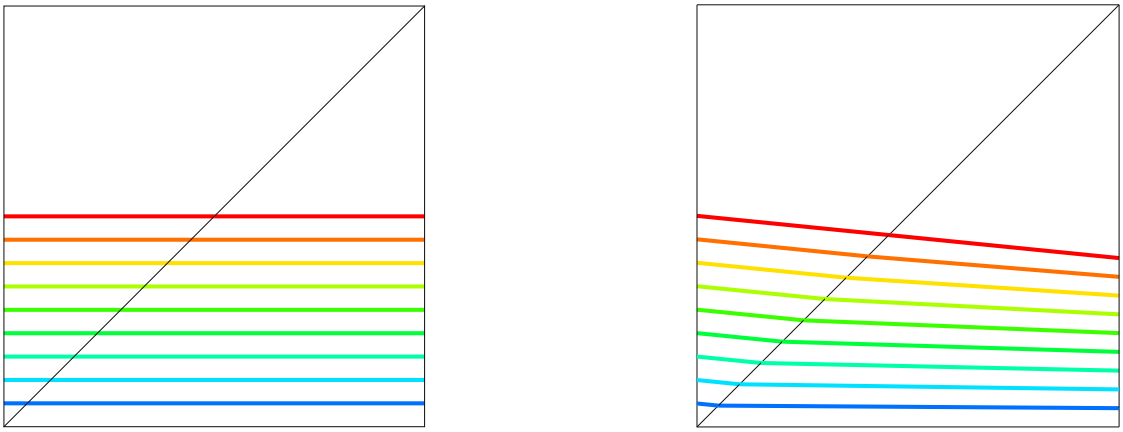
where $N_i^*$ are new shape functions taking the enrichment into account. If domain $\Omega_a$ is a perfect conductor, we retrieve the shape functions of the triangle:

$$\begin{cases} N_1^* = s \\ N_2^* = t \\ N_3^* = 1 - s - t \end{cases} \quad \text{in } \Omega_b$$

and if domain $b$ is a conductor, the shape functions become:

$$\begin{cases} N_1^* = (1 - s)(1 - t)/4 \\ N_2^* = (1 + s)(1 - t)/4 \\ N_3^* = (1 + t)/2 \end{cases} \quad \text{in } \Omega_a$$

These new conductive extended finite elements are denoted here by C-XFEM. The potential on the reference problem presented previously is illustrated in Figure 4.46. In that case the prescribed voltage on the interface is strongly prescribed even if the electrodes and the interface are not parallel.



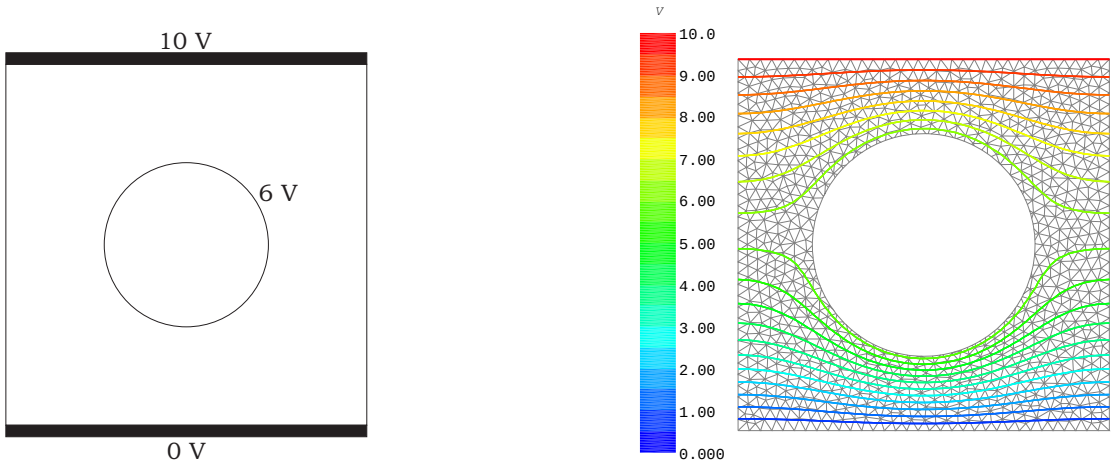
(a) Two parallel electrodes

(b) Non parallel electrodes

Figure 4.46: Imposed potential when the electrodes are parallel and oblique.

### Comparison between X-FEM and C-XFEM

The two methods are compared on the problem defined in Figure 4.47. In this example the potential difference is imposed between the upper and the lower plate. The structure defined by the Level Set is now conductive and set to 6 Volts. The domain is meshed with linear triangular elements and the solution is represented in Figure 4.47 (b).




(a) Geometry and boundary conditions

(b) Iso contours of electric potential

Figure 4.47: Reference problem.

With the C-XFEM, the boundary condition is embedded in the discretization. The only parameter influencing the model is the mesh refinement $h$. The Figure 4.48 compares the electrostatic energy of the C-XFEM element and a material-void X-FEM (noted H-XFEM) with a penalty approach to prescribe the boundary conditions.

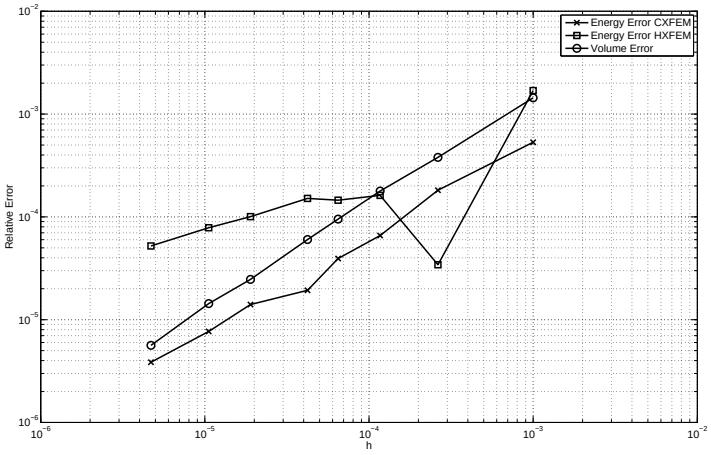



Figure 4.48: Total electrostatic energy for different mesh refinement $h$

The relative error on the total energy is unstable with respect to the solution found with FEM. When the element size is small enough, the total energy is retrieved for both models but the rate of convergence of the C-XFEM is higher. The relative errors are computed using:

$$\varepsilon_E = \frac{\|E_{XFEM} - E_{FEM}\|}{\|E_{FEM}\|}$$

where $E_{FEM}$ is the converged solution of the equivalent FEM model.

Chapter 5

# Structural optimization using X-FEM

## 5.1 Optimization methods

Three main strategies have been developed over the years to perform the optimization of a structure. The first method is the Optimality Criteria method (OC) that solves the problem with an iterative scheme based on the optimality condition characterizing the optimal solution. The second approach, the gradient method, modifies the model using an iterative procedure progressing along a descent direction calculated using gradient information. The third method refers to the sequential convex programming approach where the evolution of the structure is driven by optimization algorithm applied on a convex approximation of the structural responses.

### 5.1.1 Optimality criteria method

If one has to minimize an unconstrained analytical function, the optimality conditions require that the first derivatives of the function vanish. That is, this problem can be solved by expressing the optimality conditions in order to obtain the optimal value of the design variables.

The Optimality Criteria (OC) methods can be regarded as an extension of this approach where the solution is obtained by expressing the state conditions that characterize the optimum design and by developing efficient iterative techniques to find the optimal solution. Practically, this method can be divided in three steps. First, one derives an expression of the optimality criteria. This one can be rigorous when it is derived from a mathematical statement such as the Kuhn-Tucker conditions. It can also be intuitive as the fully stressed design (FSD) [80] criterion. The second steps aims at defining the iterative procedure that modifies the variables during the optimization. Here also, rigorous mathematical methods or heuristic procedures can be implemented to achieve the satisfaction of the optimality criteria. The last step is the definition of the iterative procedure for the Lagrange multipliers associated to the constraints.

Before the advent of the mathematical programming methods, the optimality criteria approach was the usual technique for the structural optimization. At this time, the FSD was already intensively used in practice even if they were no rigorous proof that it leads to an optimal design. The first rigorous optimality criteria method came later and is due to the work of Prager and Taylor [139] in 1968.

It is interesting to notice that the optimality criteria methods is not accepted by the whole the optimization community while mathematical programming is now generally well accepted in many fields of engineering, science, and management science. This is one of the reason why the optimality criteria method has been used principally in the structural optimization domain. Even in this domain, the acceptance of Optimality Criteria has been rude due to the lack of formal methodology of this method at its inception. The proof of usefulness of optimality criteria has been shown by Fleury [64] who demonstrated that there is a relationship between optimality criteria and dual solution methods and approximation techniques. One of the consequence of the relation established by Fleury is that Optimality Criteria are powerful tool when the number of design variables is much higher than the number of constraints.

### 5.1.2 Gradient method

The second method, the gradient method is generally applied in Level Set based optimization [8, 176]. The idea of this method is to evolve from a given solution at step $k$ characterized by the parameters $\mathbf{x}_k$ following the iterative procedure : $\mathbf{x}_{k+1} = \mathbf{x}_k + h_k \mathbf{d}_k$ where $h_k$ is a small step length and $\mathbf{d}_k$ is the descent (or ascent) direction depending the gradient of the objective function $F^{'}(\mathbf{x}_k)$.

Most structural optimization problems are generally constrained. For example one can bound the admissible volume or weight of the structure. To impose this constraint, a Lagrange multipliers method can be chosen. The descent direction is then modified to take into account the presence of the constraint by computing the derivative of the Lagrangian function associated to the problem: $\ell(\mathbf{x}_k, \lambda) = F(\mathbf{x}_k) + \lambda V(\mathbf{x}_k)$ where $V(\mathbf{x}_k)$ is the volume at step $\mathbf{x}_k$.

The choice of the step length $h_k$ must be made carefully as a too large advancing step can lead to unstable processes while a too small step can reduce dramatically the rate of convergence. Generally, the step is adapted by comparing the values of the gradient at the current and previous steps. The stopping criterion in this method can be based on the value of the gradient, the modification of the design variables or the objective function change.

### 5.1.3 Mathematical programming methods

At the beginning of the sixties, Schmit [148] and Fox [68] developed the first basis of structural optimization based on the mathematical programming (MP). Generally, in structural analysis, the optimization problem is non linear and implicit with respect to the design variables. Therefore, to obtain an optimal solution in an affordable time, Fleury [62], Schmit and co-workers [149, 150], suggested to proceed by resorting to a sequence of approximated sub-problems. Hence, the real problem is replaced by a sequence of approximated optimization sub-problem, explicit in the design variables, that can be solved effectively using a mathematical programming algorithms. Each structural response is replaced by an explicit approximation, which is generally built to be convex and separable.

MP methods and gradient-based method are built on radically different approaches from Optimality Criteria methods. Gradient and MP methods concentrate on obtaining information from the current design point in order to find the answer to two questions: in which direction should the design variables change to minimize the objective function and how much they should be modified to get the best reduction of the objective function while satisfying the constraints.

This operation is repeated until no gain on the objective can be achieved, or when the variables remains stationary.

## 5.2 General optimization problem formulation

Mathematically, a structural optimization problem $P$ is generally stated in the following general form:

$$(P) \quad \begin{cases} \min\limits_{\mathbf{x}} & g_0(\mathbf{x}) & \\ \text{s.t.:} & g_j(\mathbf{x}) \leq \overline{g_j} & j = 1 \dots m \\ & \underline{x}_i \leq x_i \leq \overline{x}_i & i = 1 \dots n \end{cases} \tag{5.1}$$

where $\mathbf{x}$ denotes the vector of variables $x_i$ which are the design parameters of the problem. In shape optimization, the design variables are related to geometrical quantities such as the radius of a circle, the position of NURBS curve control point whereas in topology optimization or in sizing, the design variables are respectively the elementary densities and an element thickness for instance. The $n$ variables $x_i$ are subject to side constraints $\underline{x}_i$ and $\overline{x}_i$ corresponding to the admissible design domain for the different variables coming from physical limitations or manufacturability considerations. The objective function, noted $g_0$, is generally a non linear implicit function that can represent the stresses, the mass, the compliance or the displacement of the structure... The inequalities defined with the $g_j(\mathbf{x})$ functions represent the $m$ design restrictions on the structural behavior of the problem. Practical constraints are usually related to the stresses of the structure, the mass or specific eigenfrequency range... The bound is noted $\overline{g}_j$. These functions are generally non-linear and non explicit with respect to the design variables. They limit the region of the design domain to be feasible design points as shown in Figure 5.1.

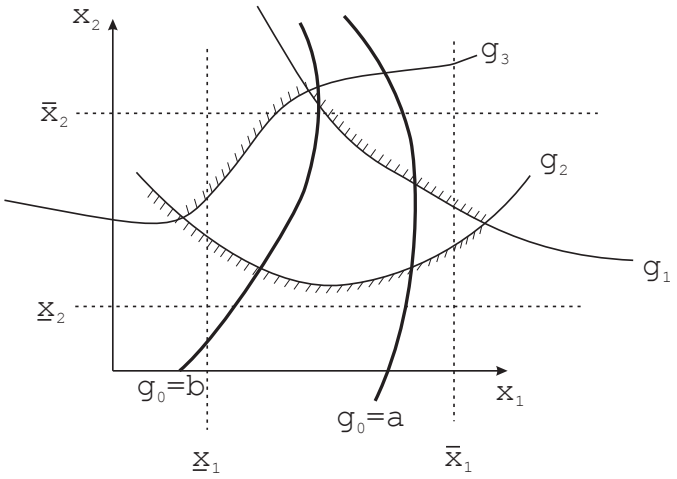



Figure 5.1: Representation of a general optimization problem

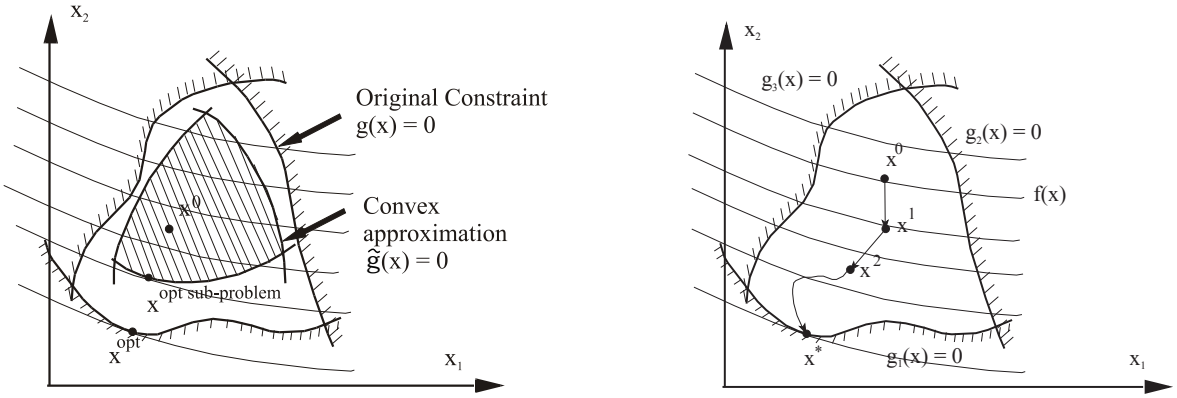
In shape optimization, the number of design variables $n$ remains usually small whereas the number of constraints $m$ can be quite large when treating problems with stress constraints, multiple load cases or multiple eigenfrequencies. In topology optimization the situation is identical to sizing but the number of variables $n$ can be very high as it generally corresponds to the number of elements.

In the most general case, all the functions involved in (5.1) are non linear and implicit with respect to the design variables. Moreover, the evaluation of these functions and their derivatives can be very 'expensive' as they require the solution of the structural finite element problem. Following the idea proposed by Fleury [62], the solution of the implicit optimization problem (5.1) can be replaced by a sequence of simpler sub-problems having a simple explicit algebraic structure in the design variables. The original optimization problem $P$ is then solved iteratively by solving the successive explicit approximated sub-problems $\tilde{P}$. The idea being to generate a sequence of sub-problems for which the successive solutions $\mathbf{x}^{k*}$ obtained forms a sequence of optimal design points that steadily improves the solution of the real problem $P$ and converges to the real solution. Mathematically, for each iteration $k$ of the optimization problem, the expression of the approximated sub-problem is stated as:

$$(\tilde{P}^k) \quad \begin{cases} \min\limits_{\mathbf{x}^k} & \tilde{g_0}^k(\mathbf{x}^k) & \\ \text{s.t.:} & \tilde{g_j}^k(\mathbf{x}) \leq \overline{g}_j & j = 1 \dots m \\ & \underline{x}_i - \alpha_i^k \leq x_i \leq \overline{x}_i + \beta_i^k & j = 1 \dots n \end{cases}$$

where $\tilde{g_j}^k (j = 0...m)$ are the explicit approximations of the ${g_j}^k (j = 0...m)$ functions around the current design point $\mathbf{x}^k$. The parameters $\alpha_i^k$ and $\beta_i^k$ are move limits introduced to control the variations of the current design point $\mathbf{x}^k$ to a certain neighborhood.

This approach has been generalized with the notion of *sequential convex programming* (SCP) [67] where each function is replaced with a convex approximation. The problem $\tilde{P}^k$ is schematically represented in Figure 5.2 where the current approximations of constraints and the objective function are built around the current design point $\mathbf{x}^k$. Given the value of the responses at the current design point, the first order derivatives and sometimes the second order derivatives, the approximated functions $\tilde{g}$ are constructed using a kind of Taylor series expansion. Then the serie of sub-problems are solved iteratively to find the next design point until convergence. Hopefully, this sequence of sub-problems yield to an optimal solution $\mathbf{x}^{k*}$ towards the real optimal point $\mathbf{x}^*$.




(a) Convex approximation of problem $P$ at iteration $k$

(b) Convergence of the successive sub-problem solutions

Figure 5.2: Convex approximation

The optimization problem can be summarized with this following steps:

1. Perform a Finite Element analysis for the given design point $\mathbf{x}^k$;

2. Compute the sensitivity analysis for the objective functions and the constraints;

3. Form the approximate optimization sub-problem given the information obtained at step 2 and 3;

4. Solve the explicit (convex and separable) approximated model with efficient mathematical programming methods;

5. Update the model with the new parameters obtained from the solution of the approximated sub-problem;

6. Verify if the stopping criterion is satisfied. If not, the procedure is restarted from the step 2 given the new design variables at step $k + 1$.

To succeed with such techniques, it is very important to obtain a high quality approximation $\tilde{g}$ without being to expensive to compute. Over the years, several approaches have been proposed to solve efficiently the $SCP$ problem. The most important step to achieve an efficient method has been suggested by Fleury [63] when he proposed to resort to a dual approach. The dual method consists in replacing the constrained primal problem with a high number of variables by its dual problem of lower dimension. In the dual space of the Lagrange multipliers, the problem is transformed into a quasi-unconstrained maximization of the dual function. The power and effectiveness of the method is directly related to the fact the maximization of the Lagrangian is realized in a sub space, which dimension is limited to the set of active constraints. Hence, this approach is very effective when the number of active constraints is much lower than the number of design variables. Moreover, if we suppose that the primal problem is convex and separable, the dual approach appears really attractive and powerful as the computation of the dual function is very effective. When these conditions hold, the $n$-dimensional Lagrangian problem can be broken into $n$ one-dimensional problems. Over the years, this approach have been the subject of a continuous improvement to obtain high quality approximations for the functions combined to effective SQP and dual approach [66] for solving the sub-problems.

## 5.3 Sensitivity analysis

In structural optimization the sensitivity analysis is the computation of the derivatives of structural response (stresses, displacements, compliance, eigenfrequency ...) with respect to the design variables. These sensitivities are the necessary ingredients to provide to the optimization algorithm in order to evaluate an effective descent direction to modify the model from an initial structure (design point) to a better (optimal) structure and build accurate mathematical approximations of the problem.

Four main approaches can be implemented to compute the derivatives of structural responses with respect to design parameters: finite differences, continuum derivatives, discrete derivatives or semi-analytical approach and automatic computer code differentiation.

### 5.3.1 Finite differences

The most simple method to compute the sensitivities is to proceed with a finite difference method. This approach consists of repeated executions of the analysis code and of the use of a finite difference method over the structural responses to obtain the derivatives. In practice, forward (5.2)

or backward differences are the most common approaches while central differences (5.3) or higher order difference schemes are extremely rare.

$$\frac{dg(\mathbf{x})}{dx_i} \approx \frac{\Delta g}{\Delta x_i} = \frac{g(\mathbf{x}+\Delta\mathbf{x}^{(i)}) - g(\mathbf{x})}{\Delta x_i} \tag{5.2}$$

$$\frac{dg(\mathbf{x})}{dx_i} \approx \frac{\Delta g}{\Delta x_i} = \frac{g(\mathbf{x}+\Delta\mathbf{x}^{(i)}) - g(\mathbf{x}-\Delta\mathbf{x}^{(i)})}{2\Delta x_i} \tag{5.3}$$

$$\text{with } \Delta\mathbf{x}^{(i)} = (0 \dots \Delta x_i \dots 0)^T$$

When the number of design variables is large, while very simple, this finite difference procedure increases dramatically the number of analysis which can be very costly. Hence, to obtain the derivative of a response with respect to a design parameter $x_i$, the initial model is perturbed by a very small quantity $\Delta x_i$ and the structural responses are recomputed for this perturbed design. Obtaining the sensitivities for a model with $n$ variables requires therefore $(n{+}1)$ complete reanalysis of the whole simulation per iteration for a forward (backward) scheme and $(2n{+}1)$ for a central scheme.

Several approaches have been developed to limit the amount of computation time associated to the number of additional analysis required to obtain the sensitivities by finite differences. The first approach consists in reducing the time of the analysis. For instance, one can simply choose to increase the different assumptions realized on the model or to reduce the size of the model, which can lead to a lower computational time. The cost reduction is then obtained by accepting a less accurate solution for the model. In certain cases, the reanalysis time can be reduced without loosing accuracy by avoiding any additional computation in regions that are not affected by the optimization process. The model is then divided into constant and modifiable regions where a super-element technique can be used.

The second approach aims at improving the computation of the derivatives by approximating the solution of the perturbed structure. Depending on the type of solver used for the structural problem, several approaches have been developed to reduce the computational effort and to obtain the solution of the perturbed structure from the solution of the nominal structure.

Hence, when direct solvers are used, several methods can be implemented to provide a fast reanalysis of the perturbed structure from the original factored stiffness matrix $K$. Unfortunately these types of technique generally suffer from inaccuracies. Among others, we can mention the Sherman-Morisson-Woodbury formula [6], which is exact when the modification is limited to a low rank modification of $K$ (if one single element is modified for instance). When the perturbation is more important, it is possible to use binomial series approximation of the inverse of $K$ (see Akgün *et al.* [6] for a discussion of several approaches). With iterative solvers, it may be possible to use the nominal solution as a good starting point for an iteration procedure to obtain the solution of the perturbed structure.

Finite difference methods can suffer from two types of errors: truncation errors and condition errors. The truncation error occurs with large perturbation step as the assumption of small perturbation is not valid anymore. Hence, the truncation error can be evaluated by writing the Taylor expansion of the function $g$:

$$g(\mathbf{x}+\Delta\mathbf{x}^{(i)}) = g(\mathbf{x}) + \frac{\Delta g}{\Delta x_i}\Delta x_i + \frac{\Delta x_i^2}{2}\frac{d^2}{\Delta x_i^2} g(\mathbf{x}+\zeta\Delta\mathbf{x}^{(i)}) \quad 0 \le \zeta \le 1$$

It comes that the truncation error $e_T$ of forward or backward difference schemes are:

$$e_T(\Delta x_i) = \frac{\Delta x_i}{2} \frac{d^2}{\Delta x_i^2} g(\mathbf{x} + \zeta \Delta \mathbf{x}^{(i)}) \quad 0 \leq \zeta \leq 1$$

and for a central scheme, one has:

$$e_T(\Delta x_i) = \frac{\Delta x_i^2}{6} \frac{d^3}{\Delta x_i^3} g(\mathbf{x} + \zeta \Delta \mathbf{x}^{(i)}) \quad 0 \leq \zeta \leq 1$$

The condition error is related to the difference between the numerical evaluation of the function and its exact value. This type of error can become very important if iterative methods are used to obtain the response functions. In this case, the choice of the initial guess of the solution can be very important in order to reduce the effect of the condition error on the derivative computation. Another contribution of the condition error is the round-off error. Small step sizes can involve computational inaccuracies associated to arithmetic errors with finite numbers of digits. Unfortunately, this type of error grows inversely with the perturbation step size, which introduces a dilemma in the choice of the perturbation step between truncation error and condition error. The influence of the step size is illustrated in Figure 5.3 [81].

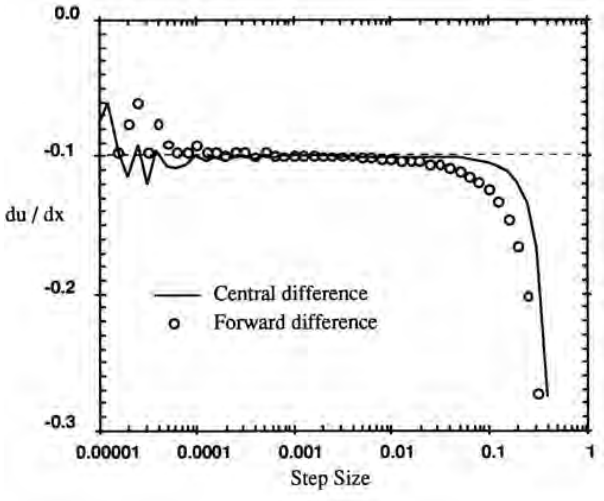



Figure 5.3: Effect of step size on derivative [81]

Finally, the remeshing of the structural domain that occurs when generating the perturbed state can also introduce a numerical noise that leads to inaccurate sensitivities.

### 5.3.2 Continuum derivatives

Continuum derivatives are obtained by differentiating the partial differential equations ruling the structural problem. From this differentiation, one obtains a set of continuum sensitivity equations, which are then solved numerically usually using the same discretization as for the structural analysis. In a static case, the equilibrium equation of a body in an initial configuration $\Omega$ is determined by the principle of virtual work:

$$\begin{aligned} a(u, \delta u) \equiv \int_\Omega D_k u_l H_{ijkl} D_i(\delta u_j) d\Omega = \\ \int_\Omega \overline{X_i} \delta u_i d\Omega + \int_{\Gamma_\sigma} \overline{t_i} \delta u_i d\Gamma \equiv l(\delta u) \end{aligned} \tag{5.4}$$

where $a(u, \delta u)$ is the bilinear energy form, $u$ the displacement field and $\delta u$ a kinematically admissible virtual displacement field. $H_{ijkl}$ is the elasticity tensor, $\overline{X_i}$ the body forces, $\overline{t_i}$ the surface traction and $\Gamma$ the boundary of $\Omega$. This principle can be applied if the displacement field satisfies a priori the compatibility requirements (5.5), the essential boundary conditions (5.6) and the constitutive relations (5.7):

$$\epsilon_{ij} = D_i u_j \quad in \quad \Omega \tag{5.5}$$

$$u_i = \overline{u}_i \quad on \quad \Gamma_u \tag{5.6}$$

$$\sigma_{ij} = H_{ijkl}\epsilon_{kl} \quad in \quad \Omega \tag{5.7}$$

where $\Gamma_u$ is the part of $\Gamma$ where displacements are prescribed.

**Sizing parameters**

When the parameters does not modify the shape of the structure, the sensitivity of the state variables can be easily obtained by taking the derivative of (5.4) with respect to the design parameters. One gets the following continuum sensitivity equation:

$$\int_\Omega \epsilon'_{kl} H_{ijkl} \delta\epsilon_{ij} d\Omega = \int_\Omega \overline{X}'_i \delta u_i d\Omega + \int_{\Gamma_\sigma} \overline{t}'_i \delta u_i d\Gamma - \int_\Omega \epsilon_{kl} H'_{ijkl} \delta\epsilon_{ij} d\Omega \tag{5.8}$$

where the superscript $'$ denotes a partial derivative with respect to the design parameter. The left-hand side of (5.8) is the same as that of (5.4) if $\epsilon$ is replaced by $\epsilon'$. Hence, the right-hand side of (5.8) defines a *pseudo-load* (or *fictitious-load*), which explicitly depends on the design. Solving the sensitivity equation (5.8) is equivalent to solving the original structural equilibrium equation (5.4), with different load cases.

**Shape parameters**

In shape optimization problems, the parameters that define the boundary curve or surface are the design variables. As these variables change, the structural boundary and the domain are modified. Therefore, when the design variables affect the shape of the domain, the differentiation of the equations of equilibrium is more complicated because the integrals depends on the design variables. Two approaches are available to obtain the sensitivity in this case. The first one consists in using the material derivative approach from continuum mechanics and the second one is based on the expression of the virtual work principle in the perturbed configuration followed by a first order Taylor expansion.

To introduce the sensitivity with respect to a shape modification of the domain, it is convenient to consider the shape of the initial structure $\Omega$ as a continuous domain and the corresponding domain $\Omega_\tau$ at fictitious times $\tau$ in the modified configuration as depicted in Fig. 5.4.
This shape modification can be regarded as a continuous deformation from the initial configuration $\Omega$ to $\Omega_\tau$. Considering that the modification of the geometry is smooth and continuous, denoting by $x$ a material point in the initial configuration and $x^\tau$ the same point in the perturbed configuration, the shape modification can be defined by a linear mapping $T$ from $\Omega$ to $\Omega^\tau$ as:

$$T \equiv x^\tau = x + \tau V(x), \quad x \in \Omega \tag{5.9}$$

in which appears the so-called *velocity field* $V(x)$ that describes the first order motion of the coordinates of a material point subject to a modification of shape design variable $\tau$:

$$V_i = \frac{\partial x_i}{\partial \tau}$$

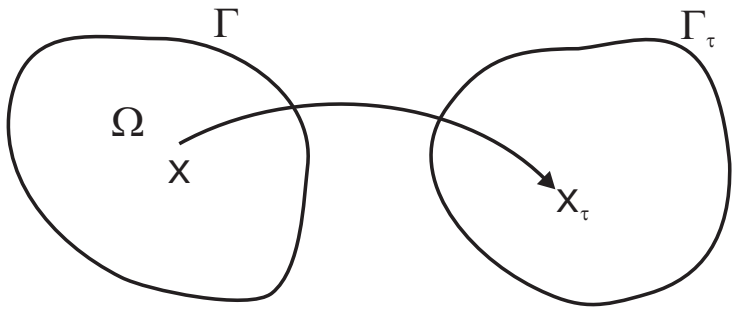



Figure 5.4: Initial and perturbed body configuration

In (5.9), the parameter $\tau$ controls the transformation between the initial and perturbed configuration. This scalar parameter is physically similar to a pseudo time controlling the amount of modification of the domain $\Omega$ and justifies the name of velocity field for $V$.

***Neighborhood transformation***

In this approach, the goal is to obtain the variational formulation verified by the derivative of the state variable $u$ of the elasticity problem with respect to a shape modification. In order to obtain the continuum sensitivity equation, (5.4) is rewritten for the perturbed configuration of the body. Omitting surface tractions ($\overline{t_i} = 0$) for convenience and considering that $\overline{X_i}$ is independent of the configuration, we have:

$$a(u^\tau, \delta u^\tau) \equiv \int_{\Omega^\tau} D_k^\tau u_l^\tau H_{ijkl} D_i^\tau(\delta u_j^\tau) d\Omega^\tau = \int_{\Omega^\tau} \overline{X_i} \delta u_i^\tau d\Omega^\tau \equiv l(\delta u^\tau) \tag{5.10}$$

Considering that the field $V(x)$ is differentiable and that one point in $\Omega$ is mapped to a unique point in $\Omega^\tau$, the neighborhood transformation is defined by:

$$\begin{aligned} dx_i^\tau &= \frac{\partial x_i^\tau}{\partial x_j} dx_j = D_j x_i^\tau dx_j, \quad i,j = 1,2,3 \\ &= D_j(x_i + \tau V_i)\, dx_j \\ &= (\delta_{ij} + \tau D_j V_i)\, dx_j \end{aligned} \tag{5.11}$$

with $D_j = \partial/\partial x_j$. The Jacobian measuring the relation between an elementary volume in $\Omega$ and $\Omega^\tau$ around the same material point, it can be easily expressed as:

$$J = [D_j x_i^\tau] = \delta_{ij} + \tau \frac{\partial V_i}{\partial x_j}$$

Or, rewritten (5.11) in matrix form:

$$dx^\tau = (I + \tau DV) dx = J\, dx$$

Inverting this relation to obtain $J^{-1}$, using Taylor expansion $1/(1+\epsilon) = 1 - \epsilon + O(\epsilon^2)$ if $\epsilon << 1$ which is also valid in matrix algebra, we have, given that $\tau$ remains small:

$$\begin{aligned} J^{-1} &= [I - \tau DV] + O(\tau^2) \\ dx_i &= (\delta_{ij} - \tau D_j V_i + O(\tau^2))\, dx_j^\tau \end{aligned}$$

The variation of an elementary volume around the material point $x$ is then given by:

$$d\Omega^\tau = det\,|J|\,d\Omega = (1 + \tau D_i V_i + O(\tau^2))\,d\Omega$$

Denoting $\tilde{u}(x) = u^\tau(x^\tau) = u^\tau(x+\tau V(x))$, the displacements in the perturbed geometry, it comes for a small variation of $\tau$:

$$D_k^\tau u_l^\tau = D_k^\tau x_m D_m \tilde{u}_l = \left[\delta_{km} - \tau D_k V_m + O(\tau^2)\right] D_m \tilde{u}_l \tag{5.12}$$

$$D_i^\tau (\delta u_j^\tau) = D_i^\tau x_n D_n (\delta u_j^\tau) = \left[\delta_{in} - \tau D_i V_n + O(\tau^2)\right] D_n (\delta \tilde{u}_j) \tag{5.13}$$

By expressing the displacement of the perturbed configuration and using the linearized expression of the Jacobian determinant:

$$\tilde{u}(x) = u(x) + \tau \dot{u}(x) + O(\tau^2) \tag{5.14}$$

where $\dot{u}=\frac{du}{d\tau}$ is the shape derivative of $u(x)$. Introducing relations (5.12), (5.13) into (5.4), neglecting all second order terms in $\tau$, we can choose $\delta\tilde{u}(x) = \delta u(x)$ because the kinematically admissible displacements are arbitrary to obtain:

$$\begin{aligned}&\int_\Omega \left[(\delta_{km} - \tau D_k V_m) D_m \tilde{u}_l H_{ijkl} (\delta_{in} - \tau D_i V_n) D_n (\delta \tilde{u}_j)\right](1 + \tau D_i V_i)\,d\Omega \\ &- \int_\Omega \overline{X}_i (1 + \tau D_p V_p) \delta \tilde{u}_i\,d\Omega = 0\end{aligned}$$

By rearranging the different terms, we have:

$$\begin{aligned}&\int_\Omega D_k \tilde{u}_l H_{ijkl} D_i(\delta u_j) + \tau [D_k \tilde{u}_l H_{ijkl} D_i(\delta u_j) D_p V_p - D_k V_m D_m \tilde{u}_l H_{ijkl} D_i(\delta u_j) \\ &- D_k \tilde{u}_l H_{ijkl} D_i V_n D_n(\delta u_j)]\,d\Omega - \int_\Omega \overline{X}_i (1 + \tau D_p V_p) \delta u_i\,d\Omega = 0\end{aligned} \tag{5.15}$$

By introducing the relation (5.14) into (5.15), we obtain:

$$\begin{aligned}&\int_\Omega \tau [D_k u_l H_{ijkl} D_i(\delta u_j) D_p V_p - D_k V_m D_m u_l H_{ijkl} D_i(\delta u_j) - D_k u_l H_{ijkl} D_i V_n D_n(\delta u_j) \\ &+ D_k \dot{u}_l H_{ijkl} D_i(\delta u_j)]\,d\Omega + \int_\Omega D_k u_l H_{ijkl} D_i(\delta u_j) - \int_\Omega \overline{X}_i \delta u_i d\Omega - \int_\Omega \tau \overline{X}_i D_p V_p \delta u_i\,d\Omega = 0\end{aligned} \tag{5.16}$$

For the initial configuration, the structure verifies the following variational equation if there is no tractions surface:

$$\int_\Omega D_k u_l H_{ijkl} D_i(\delta u_j) d\Omega - \int_\Omega \overline{X}_i \delta u_i d\Omega = 0 \tag{5.17}$$

We obtain finally the desired expression:

$$\begin{aligned}a(\dot{u}, \delta u) \equiv \int_\Omega D_k \dot{u}_l H_{ijkl} D_i(\delta u_j) d\Omega &= \int_\Omega D_k V_m D_m u_l H_{ijkl} D_i(\delta u_j)\,d\Omega + \\ \int_\Omega D_k u_l H_{ijkl} D_i V_m D_m(\delta u_j)\,d\Omega &+ \int_\Omega \left[\overline{X}_i \delta u_i - D_k u_l H_{ijkl} D_i(\delta u_j)\right] D_m V_m d\Omega\end{aligned} \tag{5.18}$$

***Material derivative approach***

Another way to obtain equation (5.18) is to use the material derivative approach. The derivative of the displacement at a point $x$ is dependent on the geometry in two ways:

- The displacement is evaluated in a domain that depends on the shape parameter $\tau$;

- The material point in which the displacement is evaluated is moved to a new position $x^\tau$;

Defining the material derivative of a quantity $u$:

$$\frac{d}{d\tau}u = \dot{u} = \lim_{\tau\to 0}[\frac{u^\tau(x^\tau) - u(x)}{\tau}] = \lim_{\tau\to 0}[\frac{\tilde{u}(x) - u(x)}{\tau}] \tag{5.19}$$

Expanding $\tilde{u}$ up to the first order:

$$\tilde{u} = u + \tau[\frac{\partial u}{\partial \tau} + \frac{\partial u}{\partial x}\frac{\partial x}{\partial \tau}] = u + \tau[\frac{\partial u}{\partial \tau} + V\nabla u]$$

we obtain the expression of the material derivative (or total derivative) of $u$ where the velocity field naturally appears:

$$\dot{u} = u^{'} + \nabla u V \tag{5.20}$$

where $u^{'}$ is the partial derivative of the displacement $\frac{\partial u}{\partial \tau}$ and $\nabla u V = [\partial u_i/\partial x_j]V_j$ is the convective term. The Jacobian matrix of the transformation is:

$$\begin{aligned} J = \frac{\partial x^\tau}{\partial x} &= I + \tau\frac{\partial V}{\partial x} \\ &= I + \tau\nabla V(x) \end{aligned}$$

where $I$ is the identity matrix and $\nabla V(x)$ is the Jacobian matrix of $V(x)$. The material derivative of $J$ becomes the Jacobian matrix of the velocity field as [47]:

$$\begin{aligned} \left.\frac{dJ}{d\tau}\right|_{\tau=0} &= \nabla V(x) \\ \left.\frac{d\,|J|}{d\tau}\right|_{\tau=0} &= divV(x) \\ \left.\frac{dJ^{-1}}{d\tau}\right|_{\tau=0} &= -\nabla V(x) \end{aligned}$$

Reminding the following the material derivative formula [47]:

$$F = \int_{\Omega^\tau} f^\tau(\mathbf{x}^\tau)d\Omega^\tau = \int_\Omega f^\tau(\mathbf{x} + \tau V(\mathbf{x}))\,|J|\,d\Omega$$

is given by:

$$\begin{aligned}
\left.\frac{dF}{d\tau}\right|_{\tau=0} &= \left.\frac{d}{d\tau}\int_\Omega f^\tau(\mathbf{x}+\tau V(\mathbf{x}))\,|J|\,d\Omega\right|_{\tau=0} \\
&= \int_\Omega \left(\frac{df(\mathbf{x})}{d\tau} + f(\mathbf{x})\,divV\right) d\Omega \qquad (5.21)\\
&= \int_\Omega \left(\frac{\partial f(\mathbf{x})}{\partial\tau} + \nabla f(x)^T V(\mathbf{x}) + f(\mathbf{x})\,divV\right) d\Omega \\
&= \int_\Omega \left(\frac{\partial f(\mathbf{x})}{\partial\tau} + div(f(\mathbf{x})V(\mathbf{x})\right) d\Omega \\
&= \int_\Omega \frac{\partial f(\mathbf{x})}{\partial\tau} d\Omega + \int_\Gamma f(\mathbf{x})V(\mathbf{x}).n\,d\Gamma \\
&= \int_\Omega f^{'}(\mathbf{x}) d\Omega + \int_\Gamma f(\mathbf{x})V_n\,d\Gamma \qquad (5.22)
\end{aligned}$$

And for a boundary functional, defined as an integral over $\Gamma^\tau$, one similarly obtains:

$$G = \int_{\Gamma^\tau} g_\tau(\mathbf{x}^\tau) d\Gamma^\tau \qquad (5.23)$$

$$\left.\frac{dG}{d\tau}\right|_{\tau=0} = \int_\Gamma \frac{dg}{d\tau} + [\kappa g]\,V.\mathbf{n}\,d\Gamma = \int_\Gamma \frac{\partial g}{\partial\tau} + \left(\nabla g^T.\mathbf{n} + \kappa g\right) V.\mathbf{n}\,d\Gamma$$

where $\kappa$=$div\,\mathbf{n}$ is the curvature of the boundary.

Applying directly (5.21) and to (5.10) we get:

$$\begin{aligned}
\left.\frac{d}{d\tau}a(u^\tau,\delta u^\tau)\right|_{\tau=0} &= \int_\Omega \frac{d}{d\tau}(D_k u_l)H_{ijkl}D_i(\delta u_j) + D_k u_l H_{ijkl}\frac{d}{d\tau}(D_i(\delta u_j))\,d\Omega \\
&\quad + \int_\Omega D_k u_l H_{ijkl} D_i(\delta u_j) D_p V_p\,d\Omega
\end{aligned}$$

Using the relation between the partial derivative and the material derivative in indicial notation (5.20): $u_j^{'} = \dot{u}_j - V_i D_i u_j$, and noting that the differentiation order can be interchanged between partial derivatives and the spatial derivatives we have:

$$\begin{aligned}
\left.\frac{d}{d\tau}a(u^\tau,\delta u^\tau)\right|_{\tau=0} &= \frac{\partial}{\partial\tau}(D_i u_j) + V_k D_k(D_i u_j) \\
&= D_i(\frac{\partial}{\partial\tau}u_j) + V_k D_k(D_i u_j) \\
&= D_i(\dot{u}_j - V_k D_k u_j) + V_k D_k(D_i u_j) \\
&= D_i\dot{u}_j - D_i V_k D_k u_j - V_k D_i(D_k u_j) + V_k D_k(D_i u_j) \\
&= D_i\dot{u}_j - D_i V_k D_k u_j
\end{aligned}$$

and finally:

$$\begin{aligned}
\frac{d}{d\tau}a(u^\tau,\delta u^\tau) &= \qquad (5.24)\\
&\int_\Omega D_k(\dot{u_l}-V_mD_mu_l)H_{ijkl}D_i(\delta u_j)+D_ku_lH_{ijkl}D_i(\delta\dot{u}_j-V_pD_p\delta u_j)\,d\Omega\\
&+\int_\Omega D_ku_lH_{ijkl}D_i(\delta u_j)D_pV_p\,d\Omega\\
&= \int_\Omega D_k\dot{u}_lH_{jikl}D_i(\delta u_j)-D_k(V_mD_mu_l)H_{ijkl}D_i(\delta u_j)\\
&-D_ku_lH_{jikl}D_i(V_pD_p\delta u_j)d\Omega\\
&+\int_\Omega D_ku_lH_{ijkl}D_i(\delta u_j)D_pV_p\,d\Omega+\int_\Omega D_ku_lH_{jikl}D_i(\delta\dot{u}_j)\,d\Omega\\
&= a(\dot{u},\delta u)+a^{'}(u,\delta u)+a(u,\delta\dot{u}) \qquad (5.25)
\end{aligned}$$

The first term on the right hand side represents the implicit dependence on the design through the field variable $u$, while the second term, the fictitious load, denotes the explicit dependence on the design velocity. The last term is equal to zero as the virtual displacement are arbitrary, $\delta\dot{u}$ can be chosen equal to zero. Similarly, the derivative of the load form becomes:

$$\begin{aligned}
\frac{d}{d\tau}l(\delta u^\tau)\bigg|_{\tau=0} &= \int_\Omega\delta(\dot{u_i})\overline{X_i}d\Omega+\int_\Omega\overline{X}_i\delta u_iD_pV_p\,d\Omega\\
&= \int_\Omega\overline{X}_i\delta\dot{u}_i\,d\Omega+\int_\Omega\overline{X}_i\delta u_iD_pV_p\,d\Omega\\
&= l(\delta\dot{u})+l^{'}(u,\delta u) \qquad (5.26)
\end{aligned}$$

Here also, the term $l(\delta\dot{u})$ is equal to zero if $\delta\dot{u}=0$. There is no term $l(\dot{u})$ because all applied loads are explicitly dependent on the design variable. If we do not assume that $\delta\dot{u}=0$, the following equality holds:

$$a(u,\delta\dot{u})=l(\delta\dot{u}) \qquad (5.27)$$

Putting all together, we have:

$$a(\dot{u},\delta u)=l^{'}(u,\delta u)-a^{'}(u,\delta u)$$

Distributing the terms in (5.25) and applying the divergence theorem on the boundary integral terms, we have:

$$\begin{aligned}
a(\dot{u},\delta u)=&\int_\Omega D_kV_mD_mu_lH_{ijkl}D_j(\delta u_i)\,d\Omega+\int_\Omega D_ku_lH_{ijkl}D_iV_nD_n(\delta u_j)\,d\Omega\\
&-\int_\Omega D_ku_lH_{ijkl}D_i(\delta u_j)D_pV_p\,d\Omega+\int_\Omega\overline{X}_i\delta u_iD_pV_p\,d\Omega
\end{aligned} \qquad (5.28)$$

Comparing equations (5.28) and (5.18), one can convince that both expression are identical.

The major advantage of the continuum approach is that the sensitivity formulation is independent of the discretization and the numerical schemes. The sensitivity equation obtained is well defined as long as the functions on the right-hand side are integrable over the domain or on the boundary. As soon as the continuum sensitivity equation is obtained, it can be discretized in the same

manner as the original analysis equations in order to obtain a system of matrix equations similar to the initial problem. Hence, rewriting (5.28) in matrix and discretized form, we have:

$$\delta\mathbf{u}^T \int_\Omega \mathbf{B}^T\mathbf{H}\mathbf{B}\, d\Omega\dot{\mathbf{u}} = \delta\mathbf{u}^T[\int_\Omega (\dot{\mathbf{B}}^T\mathbf{H}\mathbf{B} + \mathbf{B}^T\mathbf{H}\dot{\mathbf{B}} - \mathbf{B}^T\mathbf{H}\mathbf{B}\, divV)\, d\Omega\mathbf{u} + \int_\Omega \mathbf{N}^T\overline{X}\, divV\, d\Omega]$$

Defining:

$$\begin{aligned}\dot{\mathbf{K}} &= \int_\Omega (\mathbf{B}^T\mathbf{H}\mathbf{B}\, divV - \dot{\mathbf{B}}^T\mathbf{H}\mathbf{B} - \mathbf{B}^T\mathbf{H}\dot{\mathbf{B}}^T)\, d\Omega \\ \dot{\mathbf{f}} &= \int_\Omega \mathbf{N}^T\overline{X}\, divV\, d\Omega\end{aligned}$$

we have a discretized variational equation:

$$\delta\mathbf{u}^T\mathbf{K}\dot{\mathbf{u}} = \delta\mathbf{u}^T(\dot{\mathbf{f}} - \dot{\mathbf{K}}\mathbf{u})$$

and the related linear matrix equation to solve:

$$\mathbf{K}\dot{\mathbf{u}} = \dot{\mathbf{f}} - \dot{\mathbf{K}}\mathbf{u} \tag{5.29}$$

### 5.3.3 Discrete derivatives and the semi-analytic approach

In the previous section, the sensitivities are obtained by differentiating the continuum equation and solving numerically a discretized sensitivity equation. For discrete derivatives, the order of operations is reversed. First, the continuum equations are discretized (using a FEM approach for instance) and the differentiation with respect to the design variables is performed on the discrete set of equations. In a static case, the equilibrium equation of a body subject to a given load case $\mathbf{f}(z)$ is:

$$\mathbf{K}(z)\,\mathbf{u}(z) = \mathbf{f}(z)$$

Differentiating with respect to the design variables yields:

$$\mathbf{K}\frac{\partial\mathbf{u}}{\partial z} = \frac{\partial\mathbf{f}}{\partial z} - \frac{\partial\mathbf{K}}{\partial z}\mathbf{u} \Longleftrightarrow \frac{\partial\mathbf{u}}{\partial z} = \mathbf{K}^{-1}\mathbf{p} \tag{5.30}$$

where $\mathbf{p} = \frac{\partial\mathbf{f}}{\partial z} - \frac{\partial\mathbf{K}}{\partial z}\mathbf{u}$ is called the pseudo-load vector. This term corresponds to the load necessary to re-establish the equilibrium when the design is perturbed.

From equation (5.30) it is clear that, to obtain the sensitivities, one has to solve the same set of equations as solved for the initial problem. The only difference is that the right hand side term has changed being now the so-called pseudo-load vector $\mathbf{p}$. All the computational effort lies in the generation of this pseudo-load vector. $\mathbf{K}^{-1}$ being factorized and stored, solution of (5.30) requires only to operate backward solution.

Different approaches are available to compute the pseudo-load vector. The first method is the *analytical differentiation*. Depending on the nature of the design parameters, this approach can be more or less complex. For most sizing problems, the structural matrices can be expressed as a product of the design variable and an independent matrix $\overline{K}$. For instance:

$$\begin{aligned}\text{Bar element stiffness matrix}: \quad K &= \frac{EA}{L}\overline{K} \\ \text{Plane stress element stiffness matrix}: \quad K &= t\overline{K}\end{aligned}$$

where $A$ is a cross section, $t$ the thickness, $E$ the Young Modulus and $L$ the bar length. Hence, the derivative of a bar element with respect to its cross section is simply given by the matrix $\overline{K}$ itself.

This approach is very simple to implement when the parameters are explicitly present in the structural matrices expression. In these cases, the differentiation of structural matrices is straightforward. However, when the design parameters involve shape modification, the analytical differentiation becomes arduous and not trivial. Two different methods are available in this case to obtain the sensitivity of the different terms: analytical derivatives and finite differences.

**Analytical derivatives**

The analytical derivative of the different terms of equation (5.29) can be performed equivalently on the continuous expression or on the discrete form. However, one important remark should be taken into account when computing the analytical derivative. When the analytical expression is written in the real space $\Omega(z)$, which is a function of the shape parameter '$z$', one has to resort to a material derivative approach as the domain $\Omega$ evolves with the shape parameter '$z$'. Hence, the expression of the stiffness matrix derivative is given by:

$$K = \int_{\Omega(z)} B^T H B \; d\Omega \tag{5.31}$$

$$\frac{dK}{dz} = \int_{\Omega(z)} \frac{d}{dz}(B^T H B) \; d\Omega + \int_{\Omega(z)} B^T H B \, divV \; d\Omega \tag{5.32}$$

$$= \int_{\Omega(z)} \frac{d}{dz}(B^T) H B \; d\Omega + \int_{\Omega(z)} B^T H \frac{d}{dz} B \; d\Omega + \int_{\Omega(z)} B^T H B \, divV \; d\Omega \tag{5.33}$$

Expressing for instance one term of the matrix $\frac{d}{dz}B = \frac{d}{dz}\frac{\partial N}{\partial x}$, we find after using the chain rule:

$$\begin{aligned}
\frac{d}{dz}(\frac{\partial}{\partial x_i}N_j) &= \frac{\partial}{\partial z}(\frac{\partial}{\partial x_i}N_j) + V_k\frac{\partial}{\partial x_k}(\frac{\partial}{\partial x_i}N_j) \\
&= \frac{\partial}{\partial x_i}(\frac{\partial}{\partial z}N_j) + V_k\frac{\partial}{\partial x_k}(\frac{\partial}{\partial x_i}N_j) \\
&= \frac{\partial}{\partial x_i}(\dot{N}_j - V_k\frac{\partial}{\partial x_k}N_j) + V_k\frac{\partial}{\partial x_k}(\frac{\partial}{\partial x_i}N_j) \\
&= \frac{\partial}{\partial x_i}\dot{N}_j - \frac{\partial}{\partial x_i}V_k\frac{\partial}{\partial x_k}N_j - V_k\frac{\partial}{\partial x_i}(\frac{\partial}{\partial x_k}N_j) + V_k\frac{\partial}{\partial x_k}(\frac{\partial}{\partial x_i}N_j) \\
&= -\left(\frac{\partial}{\partial x_i}V_k\right)\left(\frac{\partial}{\partial x_k}N_j\right)
\end{aligned}$$

as the shape functions are independent of the design variables and equation (5.33) can be rewritten as:

$$\frac{dK}{dz} = \int_{\Omega(z)} \left\{-\nabla V B^T H B - B^T H \nabla V B \; d\Omega + B^T H B div V\right\} d\Omega \tag{5.34}$$

For example, if we consider a 1D linear finite element of length $L$, the stiffness matrix is:

$$K = \frac{EA}{L}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \tag{5.35}$$

with $E$ the Young modulus and $A$ the cross section of the bar. The sensitivity of the stiffness matrix for a modification of the length $L$ can be obtained analytically given (5.35):

$$\frac{dK}{dL} = -\frac{EA}{L^2}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \tag{5.36}$$

The velocity field $V$ can be chosen as:

$$V(x) = \frac{x}{L}$$

The divergence of this velocity field is:

$$divV(x) = \frac{1}{L}$$

and the derivative of the $B$ matrix can be also obtained analytically:

$$B = \left[-\frac{1}{L}\ \frac{1}{L}\right], \quad \frac{dB}{dL} = \left[\frac{1}{L^2}\ -\frac{1}{L^2}\right] = -\frac{\partial V}{\partial x}B \tag{5.37}$$

Thus, one can obtain the sensitivity of the stiffness matrix applying indifferently (5.34) or (5.33) to get an expression that is in total agreement with the analytical derivative (5.36):

$$\begin{aligned}
\frac{dK}{dL} &= EA\{\int_0^L -\frac{\partial}{\partial x}VB^TB\,dx - \int_0^L B^T\frac{\partial}{\partial x}VB\,dx + \int_0^L B^TBdivV\,dx\} \\
&= -EA\int_0^L \frac{1}{L}\begin{bmatrix} -\frac{1}{L} \\ \frac{1}{L}\end{bmatrix}\left[-\frac{1}{L}\ \frac{1}{L}\right]dx + EA\frac{1}{L}\int_0^L \begin{bmatrix} -\frac{1}{L} \\ \frac{1}{L}\end{bmatrix}\left[\frac{1}{L}\ -\frac{1}{L}\right]dx \\
&+ EA\int_0^L \begin{bmatrix} -\frac{1}{L} \\ \frac{1}{L}\end{bmatrix}\left[-\frac{1}{L}\ \frac{1}{L}\right]\frac{1}{L}\,dx \\
&= -2EA\int_0^L \begin{bmatrix} \frac{1}{L^3} & -\frac{1}{L^3} \\ -\frac{1}{L^3} & \frac{1}{L^3}\end{bmatrix}dx + EA\int_0^L \begin{bmatrix} \frac{1}{L^2} & -\frac{1}{L^2} \\ -\frac{1}{L^2} & \frac{1}{L^2}\end{bmatrix}\frac{1}{L}\,dx \\
&= -\frac{EA}{L^2}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}
\end{aligned}$$

However, in practice, the stiffness matrix is never computed in the physical space but in the referential space $(\xi, \eta)$ in order to use known shape functions and to apply fixed Gauss quadrature schemes. The first consequence of the introduction of a referential space is that the integration domain $\Omega_{ref}$ is not depending anymore on the shape of the element in the physical space. Hence, any modification of the parameter $z$ has no influence on the integration domain and the total derivative of $K$ is obtained by applying the Leibniz integral rule:

$$\begin{aligned}
K &= \int_{\Omega_{ref}} B_\xi^T H B_\xi\,|J|\,d\Omega_{ref} \\
\frac{dK}{dz} &= \int_{\Omega_{ref}} \frac{\partial}{\partial z}(B_\xi^T H B_\xi d\,|J|)d\Omega_{ref}
\end{aligned}$$

where $B_\xi$ is the derivative of shape functions in the reference space $(\xi, \eta)$. Noticing that $B_\xi = J^{-1}B$ and by using the previous relation $\frac{\partial}{\partial z}J^{-1} = -\nabla.V$ and $\frac{\partial}{\partial z}|J| = divV$, we obtain a tractable expression equivalent to the previous forms:

$$\begin{aligned}
\frac{dK}{ds} &= \int_{\Omega_{ref}} \left\{\frac{\partial}{\partial z}B_\xi^T H B_\xi\,|J| + B_\xi^T H\frac{\partial}{\partial z}B_\xi\,|J| + B_\xi^T H B_\xi divV\right\}d\Omega_{ref} \\
&= \int_{\Omega_{ref}} \left\{-\nabla V B_\xi^T H B_\xi\,|J| - B_\xi^T H\nabla V B_\xi\,|J| + B_\xi^T H B_\xi divV\right\}d\Omega_{ref}
\end{aligned} \tag{5.38}$$

**Analytical derivatives of discrete structural matrices**

In computer implementation, a discretized form of the stiffness matrix derivative can improve the efficiency of the derivatives computation. To this end, one should replace each term of the expression (5.38) by its discrete form or, equivalently, derivate analytically the discrete form of the stiffness matrix. Given the expression of the discrete stiffness matrix of a finite element:

$$K = \int_V B^T H B dV = \sum_{j=1}^{n_{gpt}} w_j B_j^T H B_j \ |J_j| \tag{5.39}$$

where the subscript $j$ denotes the value taken at Gauss point $j$. One obtains by taking the derivative with respect to a design parameter $z$ the discrete form of the stiffness derivative:

$$\frac{dK}{dz} = \sum_{j=1}^{n_{gpt}} w_j \left( \frac{\partial}{\partial z} B_j^T H B_j + B_j^T H \frac{\partial}{\partial z} B_j + B_j^T H B_j \frac{\partial}{\partial z} |J_j| \right) \tag{5.40}$$

Using a classical FEM, the modification of the parameter $z$ involves a movement of the node elements. From the definition of the matrix $B$ given in chapter 4, we can compute the derivative $\frac{\partial B_j}{\partial z}$. Splitting the expression of B for each node:

$$B = [B_1, B_2, ..., B_n]$$

where $n$ is the total number of nodes of the element. The sub matrices $B_r$ of node '$r$' is given by (in mechanical 2D problems):

$$B_r = \begin{bmatrix} N_{r,x} & 0 \\ 0 & N_{r,y} \\ N_{r,y} & N_{r,x} \end{bmatrix}$$

with $N_{r,y}$ and $N_{r,x}$ the derivative of the shape function of node $r$ with respect to $x$ and $y$. The relation between the shape functions in the physical space and in the reference space are expressed with the Jacobian $J$:

$$\begin{bmatrix} N_{r,x} \\ N_{r,y} \end{bmatrix} = J^{-1} \begin{bmatrix} N_{r,\xi} \\ N_{r,\eta} \end{bmatrix}$$

Noticing that derivative of the inverse of a matrix may be computed with the relation [61]: $d(X^{-1}) = -X^{-1} dX X^{-1}$, the derivation of this expression with respect to $z$ yields:

$$\begin{aligned} \begin{bmatrix} \frac{\partial N_{r,x}}{\partial z} \\ \frac{\partial N_{r,y}}{\partial z} \end{bmatrix} &= -J^{-1} \frac{\partial J}{\partial z} J^{-1} \begin{bmatrix} N_{r,\xi} \\ N_{r,\eta} \end{bmatrix} + J^{-1} \begin{bmatrix} \frac{\partial N_{r,\xi}}{\partial z} \\ \frac{\partial N_{r,\eta}}{\partial z} \end{bmatrix} = -J^{-1} \frac{\partial J}{\partial z} J^{-1} \begin{bmatrix} N_{r,\xi} \\ N_{r,\eta} \end{bmatrix} \\ &= -J^{-1} \frac{\partial J}{\partial a} \begin{bmatrix} N_{r,x} \\ N_{r,y} \end{bmatrix} \end{aligned}$$

because the derivative of the shape function in reference space $(\xi, \eta)$ are independent of the variable $z$. Indeed, using the chain rule differentiation, we can see that derivative of the shape functions in $(\xi, \eta)$ involves a term corresponding to the Gauss point movement that is zero:

$$\frac{\partial N_{r,\xi}(\xi_j, \eta_j)}{\partial z} = \frac{\partial N_{r,\xi}(\xi_j, \eta_j)}{\partial \xi} \frac{\partial \xi_j}{\partial z} + \frac{\partial N_{r,\xi}(\xi_j, \eta_j)}{\partial \xi} \frac{\partial \eta_j}{\partial z} = 0$$

In iso-parametric case, the expression of the derivative of the Jacobian can be easily expressed by:

$$\frac{\partial J}{\partial z} = \begin{bmatrix} \sum_{i=1}^{n} N_{i,\xi} \frac{\partial x_i}{\partial z} & \sum_{i=1}^{n} N_{i,\xi} \frac{\partial y_i}{\partial z} \\ \sum_{i=1}^{n} N_{i,\eta} \frac{\partial x_i}{\partial z} & \sum_{i=1}^{n} N_{i,\eta} \frac{\partial y_i}{\partial z} \end{bmatrix}$$

Gathering all terms, we obtain the derivatives with respect to a modification of the geometry of the element:

$$\begin{bmatrix} \frac{\partial N_{r,x}}{\partial z} \\ \frac{\partial N_{r,y}}{\partial z} \end{bmatrix} = -J^{-1} \begin{bmatrix} \sum_{i=1}^{n} N_{i,\xi} V_{ix} & \sum_{i=1}^{n} N_{i,\xi} V_{iy} \\ \sum_{i=1}^{n} N_{i,\eta} V_{ix} & \sum_{i=1}^{n} N_{i,\eta} V_{iy} \end{bmatrix} J^{-1} \begin{bmatrix} N_{r,\xi} \\ N_{r,\eta} \end{bmatrix}$$

where $V_{ix}$, the velocity field of node $i$ along the $x$ direction $\frac{\partial x_i}{\partial z}$, explicitly appears in the derivative. We obviously obtain here the expression found before in (5.18) and (5.28) with the continuous approach. The last term in equation (5.40) involves the derivative of the Jacobian determinant, which can be expressed by:

$$\frac{\partial}{\partial z} |J| = \frac{\partial}{\partial z} \begin{vmatrix} \sum_{i}^{n} N_{i,\xi} x_i & \sum_{i}^{n} N_{i,\xi} y_i \\ \sum_{i}^{n} N_{i,\eta} x_i & \sum_{i}^{n} N_{i,\eta} y_i \end{vmatrix}$$

or, explicitly:

$$\begin{aligned} \frac{\partial}{\partial z} |J| = & \left( \sum_{i}^{n} N_{i,\xi} V_{ix} \right) \left( \sum_{i}^{n} N_{i,\eta} y_i \right) + \left( \sum_{i}^{n} N_{i,\xi} x_i \right) \left( \sum_{i}^{n} N_{i,\eta} V_{iy} \right) \\ & - \left( \sum_{i}^{n} N_{i,\eta} V_{ix} \right) \left( \sum_{i}^{n} N_{i,\xi} y_i \right) - \left( \sum_{i}^{n} N_{i,\eta} x_i \right) \left( \sum_{i}^{n} N_{i,\xi} V_{iy} \right) \end{aligned}$$

In the continuous approach, the sensitivity is computed taking into account a modification of the domain of integration and the modification of the shape functions defined in the physical space. In the discrete approach, the point of view is focused on the elementary level. Therefore, there is no modification of the integration domain in $(\xi, \eta)$ nor the shape functions. The effect of shape variation relies only in the terms that comes from the modification of the Jacobian.

The major difficulty of this approach, as well as the pure analytical approach, which leads to the same expression, is that one needs to compute the velocity field over the whole domain $\Omega$. Hence, when the shape parameter '$z$' represents a geometrical characteristics such as the radius of a circle, an analytical velocity field can be chosen as: $V(x, y) = (\frac{x}{r}, \frac{y}{r})$ and used all over the domain $\Omega$. However, it is generally difficult to obtain or to use an analytical velocity field in practice. Generally, the method employed consists in:

1. Perturbing the boundaries with a small $\delta z$;
2. Compute the new positions of the boundary nodes;
3. Adapt the mesh according to the movement of the boundary nodes using a smoothing technique such as a Laplacian method for instance;

4. Compute the velocity field on the whole domain;

5. Apply discrete sensitivity analysis to obtain the derivatives.

The difficulty of this method being to adapt the mesh while keeping a constant mesh topology [36] and to compute the velocity field given a known movement on the boundary of the structure. A comprehensive review of the different methods available to realize this can be found in [183].

#### Finite difference

The most common approach to obtain the sensitivities with a discrete differentiation is the semi-analytical approach. Given the analytical derivative of the discrete system of the governing equations, the derivatives of stiffness matrices and load vectors are computed with finite differences. Generally, this leads to a procedure where the finite difference is realized at the element level whilst the analytical differentiation applies to the whole system. Hence, the elementary finite differences are assembled to build the discrete sensitivity system of equations. The pseudo-load vector is generally evaluated with finite difference schemes too. Here also, forward and backward schemes are the most usual. This type of design sensitivities is commonly referred to semi-analytical discrete design sensitivities.

We should also mention that the discrete derivative or semi-analytic approach is the most common technique and is available in several industrial finite element software codes such as NASTRAN or SAMCEF. Implementations are mainly based on analytical elementary derivatives or elementary finite differences. Compared to other techniques, the discrete method presents the advantage of being computationally effective, generic and simple. The effective characteristic comes from the fact that the system of equations to be solved is equivalent to the initial problem. Moreover, one can take advantage of keeping the factorized $K$ matrix and the restitution of displacement derivative is very cheap as only a backward elimination is necessary. This approach is generic and it reduces the cost of code maintenance. Indeed, the elementary function used to generate the stiffness matrix is able to compute the sensitivity with an elementary finite difference approach. Conversely, analytical sensitivity needs the implementation of specific elementary routines that calls for different implementations and code maintenance.

### 5.3.4 Automatic computer code differentiation

The last method to obtain the sensitivities refers to the computational differentiation. This approach works at the software level and consists in directly differentiate the computer code itself. The computational differentiation approach is based on the definition of the partial derivatives of elementary functions. Given these partial derivatives, the total derivative of complex routines can then be constructed using the chain rule differentiation.

Different computer software are available with both first and higher order derivatives (ADIFOR [32] for FORTRAN and ADOL-C [76] for C/C++) to compute the derivative of programs. Initially, it was believed that pure automatic computer code generation was possible with this approach. However practical implementations have shown that human intervention is generally necessary to obtain maintainable and efficient code.

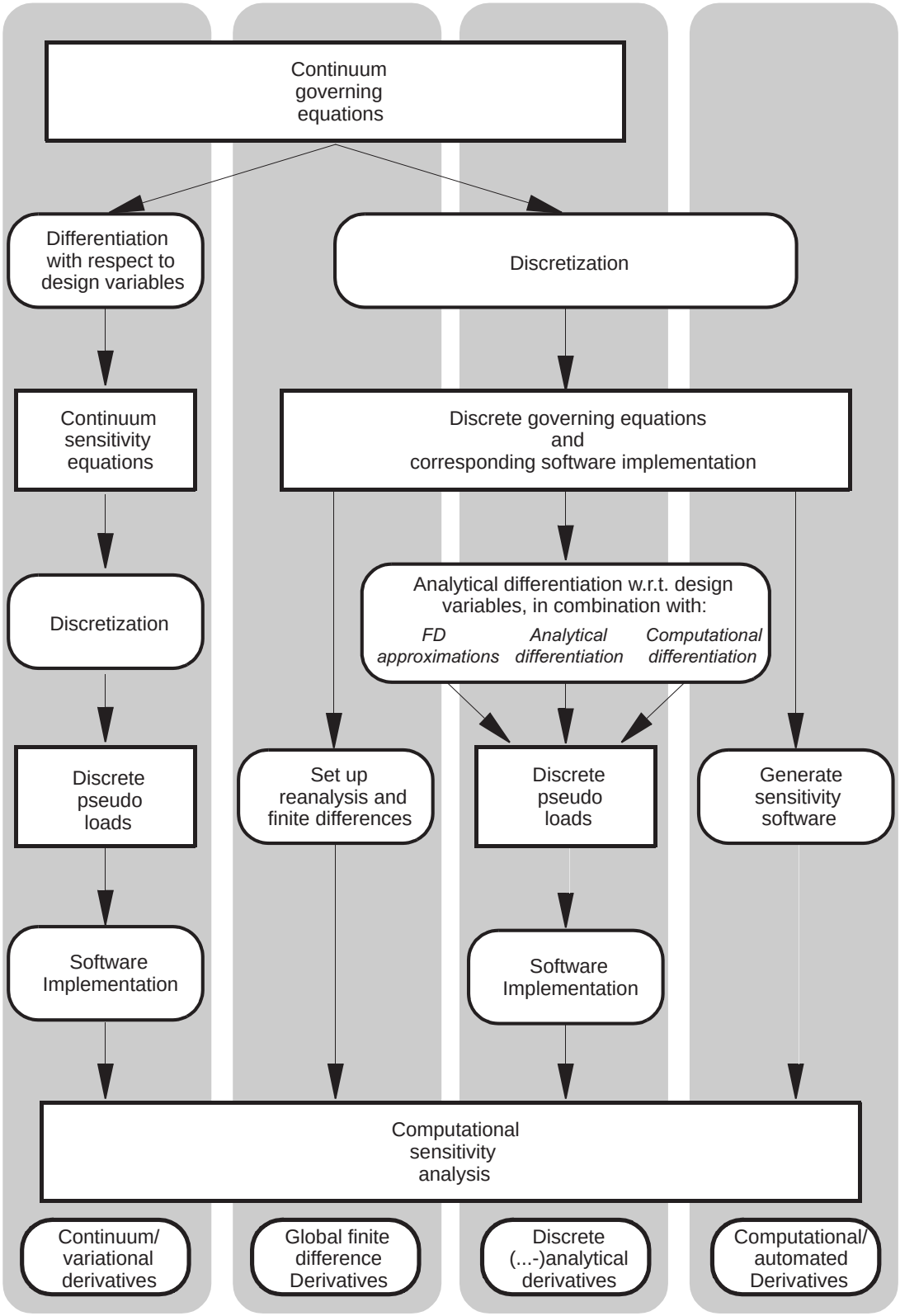



Figure 5.5: Overview of the different sensitivity analysis approaches [172]

### 5.3.5 Direct method and adjoint variable method

Two different approaches are available to compute practically the derivative of a function $g$ that represents any performance measure or constraint of the optimization problem:

$$g = g(z, u(z)) \tag{5.41}$$

where $u$ is the solution of the state equation (i.e. the displacements). This function depends on the design in two ways: first through the explicit design parameter $z$, and second, through implicit dependence on the displacement $u$. Using the chain rule differentiation, the total derivative of $g$ with respect to $z$ is given by:

$$\frac{dg}{dz} = \frac{\partial g}{\partial x} + \frac{\partial g}{\partial u}\frac{\partial u}{\partial z} \tag{5.42}$$

#### Direct differentiation method

The direct differentiation method, also known as the *pseudo load method*, evaluates the implicit dependence of $u$ with $x$ using the derivative of the structural equation:

$$\frac{\partial u}{\partial z} = K^{-1}\left[\frac{\partial f(z)}{\partial z} - \frac{\partial K(z)}{\partial z}u(z)\right] \tag{5.43}$$

Replacing the expression of $\partial u/\partial z$ into (5.42), we obtain an expression for the derivative of $g$:

$$\frac{dg}{dz} = \frac{\partial g}{\partial z} + \frac{\partial g}{\partial u}K^{-1}\left[\frac{\partial f(z)}{\partial z} - \frac{\partial K(z)}{\partial z}u(z)\right] \tag{5.44}$$

However, this expression is not very useful in practice because the term $K^{-1}$ is generally not available. Therefore, in the direct approach, (5.43) is first solved for each variables and then the sensitivity is obtained by taking the scalar product with $\frac{\partial g}{\partial u}$. This way, the solution of (5.43) should be computed once for each design variables leading to $n$ additional backward substitutions.

#### Adjoint method

When the number of variables $n$ is greater than the number of constraints $m$, it is better to adopt the *adjoint method*. To this end, a first adjoint variable vector $\Lambda$ is defined as the solution of the following system:

$$K\Lambda = \frac{\partial g}{\partial u} \tag{5.45}$$

As the mechanical problem is a self adjoint point problem, the stiffness matrix is symmetric. We can replace the expression of $\frac{\partial g}{\partial u}$ in (5.42), yielding to the following adjoint equation:

$$\frac{dg}{dz} = \frac{\partial g}{\partial z} + \Lambda^T\left(\frac{\partial f}{\partial z} - \frac{\partial K}{\partial z}u\right) \tag{5.46}$$

The adjoint method requires one backward substitution per constraint, thus $m$ backward substitutions for the whole set of constraints. The adjoint variable method is also known as the virtual load because $\frac{\partial g}{\partial u}$ acts as a dummy load. When the constraint $g$ represents an upper limit on the value of a single displacement component, the dummy load has only one non zero component corresponding to the constrained displacement component. Similarly to the direct approach, we can take advantage of using the matrix $K$, which is generally available in a factored form.

Obtaining the solution of $\partial u/\partial z$ is thus much cheaper than solving the original problem.

In practical problems with several load cases, the cost of the virtual load approach is equal to the number of active constraints $m$ and it is nearly independent of the number of load cases. Indeed, the number of critical constraints does not change significantly with the number of load cases at the end of the convergence process. With the direct method, it is necessary to introduce $n * c$ additional load vectors with $n$ the number of design variables and $c$ the real number of load cases. The number of linear systems to be solved is equal to $n_{add} = n * c$ in this case.

The direct approach seems, at first inspection, much more complex to set up. Hence, on has to divide the structural analysis in two steps. First the nominal structure has to be solved for all load cases. Second, the pseudo-load vectors are assembled and the sensitivity system is solved for all pseudo-load vectors. However, one can notice that the number of computations using this methods is preferable when the we have $n * c \leq m$. The selection between the two methods can be introduced automatically in the sensitivity analysis method. The choice is quite easy as the comparison between the number of additional load cases required for each method is known. In practice, this choice can be done at the beginning or interactively by selecting the set of (nearly) active constraints at the current design point.

## 5.4 Optimization based on the Level Set Method

Inspired by the ability of the Level Set method to modify the shape of curves and surfaces as well as allowing the modification of topology, this method has drawn a lot of attention among researchers in the field of topology optimization of mechanical structures. The first paper presenting a method related to the topic is attributed to Sethian and Wiegmann in [153]. In this study, the movement of the Level Set is driven by the Hamilton-Jacobi equation (see eq.(2.3)) and the structural analysis is carried out by an Immersed Interface Method (see [100]), which avoids the mesh generation. The optimization is based on an evolutionary approach based on the idea of removing material in regions of low stress with a removal rate, determining the velocity of the moving boundaries. Holes can be created during the optimization by modification of the Level Set based on a criteria similar to evolutionary methods. This method obtained very interesting results for different 2D structures. However, no other works has been published on this method, probably because it is based on a finite difference and not on a finite element method.

Another approach of topology optimization with Level Set has been initially proposed by Allaire *et al.* [8, 9] and closely followed by Wang *et al.* [178] who extended the method to multi-material optimization [176], thermoelastic problems [181] and maximization of simple or repeated fundamental eigenvalues of vibration [180]. In these works, the structural analysis is carried out by a finite element analysis in which the void domain is represented with a soft material similarly to a topology optimization. The sensitivity analysis is derived from material derivatives or boundary variation approach of the different cost functions and the associated augmented Lagrangian formulation of the optimization problem in case of constraints. Given the expression of the derivatives, reminding that the normal velocity field $V_n$ appears in the structural sensitivity, a proper descent direction is chosen in order to obtain a negative derivative ensuring the minimization of the objective function. The chosen velocity field $V_n$ is plugged into the Hamilton-Jacobi equation in order to drive the evolution of the Level Set and to modify the shape of the structure.

Indeed, from this point of view, this approach is quite different from a classical shape optimization. In shape optimization, the velocity field is imposed by the design variables defined for the problem and it restricts therefore the admissible set of possible domain shapes. Moreover, the intensity of the modification of design variables is retrieved from the optimization algorithm given the derivatives of the objective and cost functions. Using the Level Set approach of Allaire and Wang, the optimization formulation does not focus on determining the sensitivity of the design variable as these ones do not appear directly into the optimization problem. Instead, as a velocity field towards any direction can be constructed on the structure allowing to modify freely the design domain, the best direction of evolution (i.e. the velocity field) is chosen in order to guarantee that the objective function is diminishing. Thus, the sensitivity of the design variables is not really computed in the sense that it is only the velocity field which is computed. Given this velocity field, the modification of the design (the Level Set values) is realized by solving the Hamilton-Jacobi equation driving the Level Set with the prescribed velocity field. From this point of view, this method can be classified as a topology optimization rather than a shape optimization. The main advantage of the Level Set approach compared to a classical topology method being that the solution is composed of a black and white design without checkerboard problems. Moreover, the level of automation of this method can be considered as nearly equivalent to the one obtained in topology optimization. However, this method is not able to create naturally new holes in the structure as the sensitivity analysis is based on small perturbations of the boundary and does not provide a mechanism for nucleation of holes. In order to overcome this limitation in practice, many holes are often inserted into the initial design and allowed to merge when the boundaries evolve. A consequence of this strategy is that the result of the optimization process appears to be depend of the initial structure. To overcome the difficulty of generating new holes, Allaire *et al.* [10] proposed to combine it with the concept of topological derivatives which appears to be very promising. In [179], Wang *et al.* use a Radial Basis function Level Set method with an extension of the velocity field that allows the creation of new holes.

While this approach has proven to be very attractive and successful, it faces some limitations. Actually, all the approaches based on the Level Set Method in optimization use a Gradient Method with a Lagrangian approach. Hence, at step $k$ of the optimization process, a FEM analysis is performed, then, given the velocity $V_n$, the Hamilton-Jacobi equation is solved for a fixed number of iterations to modify the geometry of the structure. Thus, the advancing step $h$ of the gradient method, which corresponds here to the number of iterations for the Level Set equation, have to be chosen. Moreover, the CFL conditions have to be fulfilled for this resolution, limiting therefore the amplitude of the shape modification per iterations and increasing the computational time spent in the resolution of the Level Set motion equation. Thus, this difficulty increases the total computational time and leads to a high number of iterations for the global optimization problem (at least 100 iterations).

In [26], Belytschko *et al.* proposed to used the Level Set Method with a structural analysis performed using X-FEM. The design variables of the optimization are the nodal values of the Level Sets and the optimization process is solved with optimality criteria. This method is detailed for the purpose of minimizing compliance subject to a constraint on the volume but the approach could be extended to other results. The authors obtained very interesting results for different 2D structures with one or multi-material properties and the method seems to be able to create holes.

The closest method to the approach developed in present thesis is related to Chen *et al.* [42] where the analysis of the structure is realized using a Meshless approach and the sensitivity

of the objective function is based on analytical derivatives of the equilibrium equations. In this paper, the shape of the structure is represented using a combination of several Level Sets with Boolean operators. The original idea of this research comes from the fact that the authors propose to combine both parametric Level Set functions based on simple primitives characterized by a small number of parameters and discrete Level Sets (called free-form Level Set) where the parameters are the nodal Level Set values. This approach offers a great flexibility as both shape and topology optimization can be performed simultaneously. Moreover, in order to obtain a differentiable approximation when computing the sensitivity of the Level Set function, the boolean operators are transformed using R-functions [156]. The optimization procedure is based on an Augmented Lagrangian technique and the parameters evolution is realized with a gradient method. Similarly to the approaches of Allaire and Wang where the velocity field is chosen in order to guarantee a descent direction, Chen *et al.* [42] select an adapted minimizing direction for each design parameter given the expression of the sensitivity.

## 5.5 Formulation of optimization problem

In the present work, the formulation of the optimization problem is similar to the classical shape optimization problem, but its solution is greatly simplified thanks to the use of the X-FEM and the Level Set description as no mesh perturbation is needed.

The geometry and the material repartition are specified using Level Sets representations. The positive part of the Level Set represents the region where the material lies and the negative part is void. The user has a library of basic geometric features (in Level Sets) that can be combined to create almost any structural geometry. In our implementation, the available geometric features are circles, ellipses and all polygons. The design variables are chosen among the geometric parameters of these features. Beside these pre-defined Level Set shapes, the user can also build the Level Set from a NURBS curve, a surface or a general set of points. In this case, the design variables are the control points of the NURBS. Hence, each Level Set entity developed in OOFELIE has a number of given design variables that allow to modify the geometry. Moreover, using a Constructive Solid Geometry Level Set approach with the boolean operators, it is possible to build complex shapes and to proceed to a shape optimization among the design parameters of each Level Set in the model.

The optimization problem aims at finding the best shape while minimizing a given objective function and satisfying mechanical and geometrical design restrictions. The mechanical constraints can either be global responses (e.g. compliance, volume) or local criteria such as displacements or stress constraints. The number of design variables is generally small as in shape optimization. However the number of constraints may be large if local stress restrictions (e.g. stress constraints) are considered. Nonetheless, large scale problems as in topology optimization are thus avoided.

Because of the X-FEM characteristics, the geometry has not to coincide with the mesh and the shape optimization problem is carried out on a *fixed mesh*. One works here in an Eulerian approach and not in a Lagrangian approach. This circumvents the mesh perturbation problems of classical shape optimization. Sensitivity analysis does not require to extend the velocity field to the entire domain anymore. The present formulation is then, up to a certain point, simpler. However, some technical difficulties can be encountered if a finite difference or a semi-analytical scheme is used for sensitivity analysis as explained in the next section. Basically, the problem is

that the perturbation must not modify the number of degrees of freedom of the X-FEM stiffness matrix.

The Level Set approach is very convenient to modify the geometry because the Level Sets (and so the holes) can penetrate each other or disappear. Creation of new holes is more problematic since it leads to a non smooth problem. Topological derivatives (see [73, 128, 160]) have to be used for a rigorous treatment of the problem. This capability has not been studied in this thesis and is let as an extension of the method for future works.

## 5.6 The sensitivity analysis method

In a preliminary work [174] on optimization with X-FEM and Level Sets, we performed sensitivity analysis using a simple finite difference scheme. This choice was related to the objective of the study, which was to assess the capabilities of this new approach. As promising results were obtained, it was decided to develop a more efficient method for sensitivity analysis in order to handle more complex problems.

As in classical shape optimization, the sensitivity analysis of mechanical responses (such as compliance, displacement, stress ...) is carried out using a semi-analytic approach. In this approach the derivatives of stiffness matrix ($\mathbf{K}$), mass matrix ($\mathbf{M}$) and load vectors ($\mathbf{f}$) are calculated by finite differences using a small perturbation $\delta z$ of Level Set parameters. Using forward finite differences, the expression of the derivatives of the matrices are:

$$\begin{aligned}
\frac{\partial \mathbf{K}}{\partial z} &\simeq \frac{\mathbf{K}(z+\delta z)-\mathbf{K}(z)}{\delta z} \\
\frac{\partial \mathbf{M}}{\partial z} &\simeq \frac{\mathbf{M}(z+\delta z)-\mathbf{M}(z)}{\delta z} \\
\frac{\partial \mathbf{f}}{\partial z} &\simeq \frac{\mathbf{f}(z+\delta z)-\mathbf{f}(z)}{\delta z}
\end{aligned}$$

These derivatives are then exploited to compute the sensitivity of the various objective functions.

### 5.6.1 Static analysis

In static analysis, the equation to solve is given by the following equilibrium equation:

$$\mathbf{K}\mathbf{u} = \mathbf{f}$$

#### Displacement derivative

The sensitivity of the generalized displacement is obtained by differentiating the discretized equation with respect to the design variables:

$$\frac{\partial \mathbf{u}}{\partial z} = \mathbf{K}^{-1}\left(\frac{\partial \mathbf{f}}{\partial z} - \frac{\partial \mathbf{K}}{\partial z}\mathbf{u}\right)$$

that requires $n$ backward substitutions for a problem of size $n$.

**Compliance derivative**

The compliance $C$ is defined as the work of the applied load and writes in the discretized from:

$$C = \mathbf{f^T u} = \mathbf{u^T K u}$$

It is equal to twice the strain energy. In the case of dead loading forces, the expression of the generalized displacements sensitivity allows the derivative of the compliance $C$ to be expressed as a function of the stiffness matrix derivative only:

$$\frac{\partial C}{\partial z} = -\mathbf{u}^T \frac{\partial \mathbf{K}}{\partial z}\mathbf{u}$$

If the external loads are depending on the parameter $z$, one has:

$$\begin{aligned} \frac{\partial C}{\partial z} &= \mathbf{u}^T \frac{\partial \mathbf{K}}{\partial z}\mathbf{u} + 2\mathbf{u}^T \mathbf{K}\frac{\partial \mathbf{u}}{\partial z} && (5.47) \\ &= -\mathbf{u}^T \frac{\partial \mathbf{K}}{\partial z}\mathbf{u} + 2\mathbf{u}^T \frac{\partial \mathbf{f}}{\partial z} && (5.48) \end{aligned}$$

where the expressions (5.47) and (5.48) are equivalent but (5.47) involves $n$ backward substitution whereas (5.48) does not need any substitution.

The strain energy being a global quantity of the structure that is obtained by summation over all finite elements, one can also compute the sensitivity of $C$ on each element $e$:

$$\frac{\partial C_e}{\partial z} = -\mathbf{u_e}^T \frac{\partial \mathbf{K_e}}{\partial z}\mathbf{u_e} + 2\mathbf{u_e}^T \frac{\partial \mathbf{f_e}}{\partial z}$$

**Stress derivative**

The objective function or constraint may involve the stresses so that their sensitivities are needed. Two ways are possible to get the derivative of the stresses $\sigma$:

$$\sigma = \mathbf{H}\epsilon = \mathbf{HBu} = \mathbf{Tu}$$

where $\mathbf{H}$ is the Hooke's matrix and $\mathbf{B_j}$ the matrix of the derivated shape functions of the element $e$.

The first one consists in derivate the expression of the stresses $\sigma$ in all the elements:

$$\begin{aligned} \frac{\partial \sigma}{\partial z} &= \frac{\partial \mathbf{T}}{\partial z}\mathbf{u} + \mathbf{T}\frac{\partial \mathbf{u}}{\partial z} \\ &= \frac{\partial \mathbf{T}}{\partial z}\mathbf{u} + \mathbf{T}\mathbf{K}^{-1}(\frac{\partial \mathbf{f}}{\partial z} - \frac{\partial \mathbf{K}}{\partial z}\mathbf{u}) \end{aligned}$$

This approach has the drawback of being cumbersome because one has to compute the derivative of the $T$ matrix. On one hand, this can be achieved using a finite difference approach but requires the generation of two matrices $T(z+\delta z)$ and $T(z)$ which are generally not computed in industrial codes. On the other hand, one can proceed to the analytical derivation of $T$ as suggested by Braibant [37] for FEM shape optimization that calls for the computation of the velocity field $V$ and $B$ matrix derivative:

$$\frac{\partial \sigma_e}{\partial z} = \int_{\Omega_e} \mathbf{H}\left[\mathbf{B_e}\, div\mathbf{V} - \frac{\partial \mathbf{B_e}}{\partial z}\right] d\Omega_e$$

The expression of these different terms are detailed in section 5.8 for the case of X-FEM.

The second approach is based on the computation of the stresses related to the perturbed state by using the expression of the displacement sensitivities and a Taylor expansion of the stresses. Neglecting second order terms, the stress derivative can be approximated by:

$$\begin{aligned} \sigma(z+\delta z) &= \sigma(z) + \frac{\partial \sigma}{\partial z}\delta z + O(\delta z^2) \\ \Rightarrow \frac{\partial \sigma}{\partial z} &\simeq \frac{\sigma(z+\delta z) - \sigma(z)}{\delta z} \end{aligned}$$

If we approximated the stresses for the configuration $z + \delta z$ with:

$$\sigma_{\mathbf{e}}(z+\delta z) \simeq \mathbf{HB_e}(z+\delta z)\mathbf{u}(z+\delta z)$$

we have a tractable expression for the stress derivative for element $e$:

$$\frac{\partial \sigma_e}{\partial z} \simeq \frac{\mathbf{HB_e}(z+\delta z)\mathbf{u}(z+\delta z) - \mathbf{HB_e}(z)\mathbf{u}(z)}{\delta z}$$

where $\mathbf{u}(z+\delta z)$ is also computed with a first order Taylor expansion. This procedure reduces the sensitivity of the stresses as a function of the displacement derivative. In the present study, it is this second method which has been implemented.

### 5.6.2 Modal analysis

When the structural analysis is based on the modal characteristic of the structure, the structural response is obtained by solving the eigenvalue problem:

$$\left[\mathbf{K}(z) - \omega^2 \mathbf{M}(z)\right] \mathbf{u}(z) = 0$$

where $\omega^2$ is the squared of the pulsation and $u$ is the vector of eigenmodes. Generally, the eigenmode vectors are normalized with respect to the mass matrix $M$ such that:

$$\mathbf{u}^T \mathbf{M} \mathbf{u} = 1$$

In this thesis, two types of problem are considered: structural linear elastic analysis and modal analysis. If the objective of the optimization is maximizing the eigenfrequency, we have to solve the following eigenvalue problem:

$$\left(\mathbf{K} - \omega^2 \mathbf{M}\right) \mathbf{u} = \mathbf{0}$$

where $\mathbf{M}$ is the mass matrix, $\mathbf{u}$ is the eigenmode and $\omega^2$ is the squared pulsation. To optimize the structure with respect to eigenfrequencies, it is necessary to determine their sensitivities. In the case of a single eigenfrequency, the sensitivity of the $k^{th}$ eigenfrequency with respect to an arbitrary design parameter $z$ is given by:

$$\frac{\partial \omega_k^2}{\partial z} = \mathbf{u}_k^T \left(\frac{\partial \mathbf{K}}{\partial z} - \omega_k^2 \frac{\partial \mathbf{M}}{\partial z}\right) \mathbf{u}_k \tag{5.49}$$

if we use the orthogonality property of the eigenmodes. In case of multiple eigenfrequencies, the expression (5.49) can not be used anymore. However extension to multiple eigenfrequencies is possible. See [155] for example of such treatment.

### 5.6.3 A first numerical procedure

In CAD based shape optimization, the complexity of computing the stiffness matrix sensitivity stems from the modifications of the mesh associated to the perturbation $\delta z$ and from the related velocity field calculation. In fact, the difficulty is related to the relocation of the mesh nodes inside the domain following the given perturbation of the boundary nodes. In the X-FEM based approach, one has not to deal with the mesh perturbations as one works on a fixed grid. However, this method exhibits a different drawback with respect to the FEM shape optimization as the number of solid elements may change. The critical situation happens when the number of nodes is modified and the Level Set is very close to a node (see Fig. 5.6). During the perturbation $\delta z$ of the Level Set, it is possible that some empty elements become partly filled with material and are introduced into the formulation. The new elements then introduce some new nodes and new DOFs. Therefore, the number of degrees of freedom change and the dimension of the stiffness matrix (i.e. of the problem) is modified before and after the Level Set perturbation. It is not possible to perform finite differences since the matrices have different dimensions. However, even if it is possible in theory, this situation is extremely rare for unstructured mesh used in practice.

Of course, the ultimate solution to the problem should resort to a fully analytical sensitivity of the stiffness matrix. However, this would be rather restrictive for industrial applications as this procedure has to be introduced in each element definition the semi-analytical procedure is element independent.

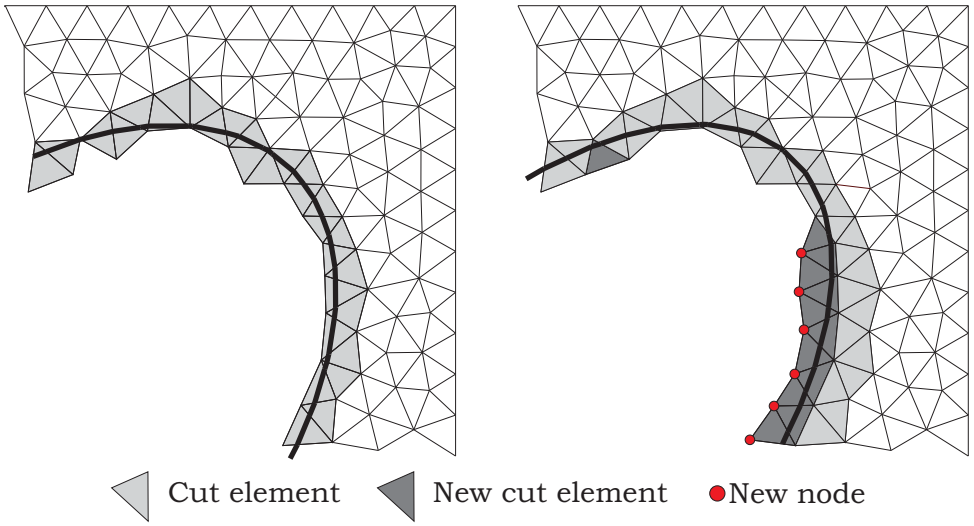



Figure 5.6: Sensitivity difficulty with semi-analytic approach

However, practically a simple procedure can be tailored. To circumvent the problem of DOF introduction, the first strategy that we have implemented is to take into account only the displacement $u_i$ for the elements that are present in the reference configuration while the contributions coming from the new partly filled elements are ignored. Hence, the number of elements is preserved and the size of the stiffness matrix remains unchanged.

### 5.6.4 Validation of the simplified approach

The plate with an elliptical hole subject to a non uniform bi-axial loading (Fig. 5.7) is a classical benchmark to validate the approximated semi-analytical sensitivity analysis. Table 5.1 gives the

sensitivities of the compliance calculated by finite differences (FD) and by a semi-analytical (SA) approach for different combinations of the design variables $a$ and $b$, i.e. the major and minor axes of the elliptical hole. The results are obtained with a relative perturbation of the design variables of $\delta = 10^{-4}$. The table 5.2 presents the comparison between semi-analytical and finite difference sensitivity analysis for the maximum Von Mises stress with respect to the parameters $a$ and $\theta$ of the ellipse.

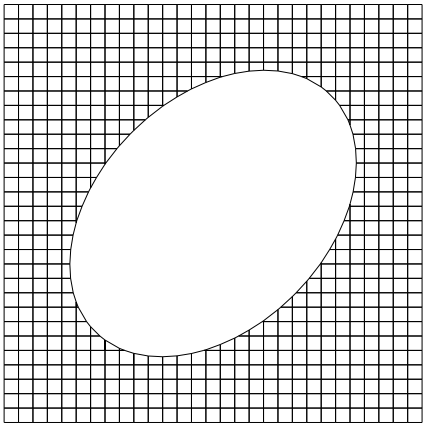


Figure 5.7: Structure used for the analytical sensitivity computation

Compliance sensitivity

| *Design variables* | *FD* | *SA* | *Relative error (%)* |
|---|---|---|---|
| $a = 0.8$ | 9083.912935 | 9079.929156 | 0.04% |
| $b = 0.55$ | 6931.355633 | 6930.048399 | 0.02% |
| $a = 0.64$ | 4229.584903 | 4228.178721 | 0.03% |
| $b = 0.44$ | 3255.032199 | 3254.570006 | 0.01% |

Table 5.1: Validation of semi-analytical sensitivity analysis approximation for compliance.

Von Mises stress sensitivity

| *Design variables* | *FD* | *SA* | *Relative error (%)* |
|---|---|---|---|
| $a = 0.6$ | $3698,0000$ | $3691,3344$ | $0,1802$ |
| $\theta = \pi/4$ | $478,0000$ | $477,0641$ | $0,1957$ |
| $a = 0.6$ | $2712,000$ | $2707,3283$ | $0,1722$ |
| $\theta = \pi/6$ | $523,7000$ | $523,4099$ | $0,0553$ |
| $a = 0.6$ | $783,8000$ | $782,3920$ | $0,1796$ |
| $\theta = 0$ | $11,6239$ | $11,6235$ | $0,0029$ |

Table 5.2: Validation of semi-analytical sensitivity analysis approximation for Von Mises stresses

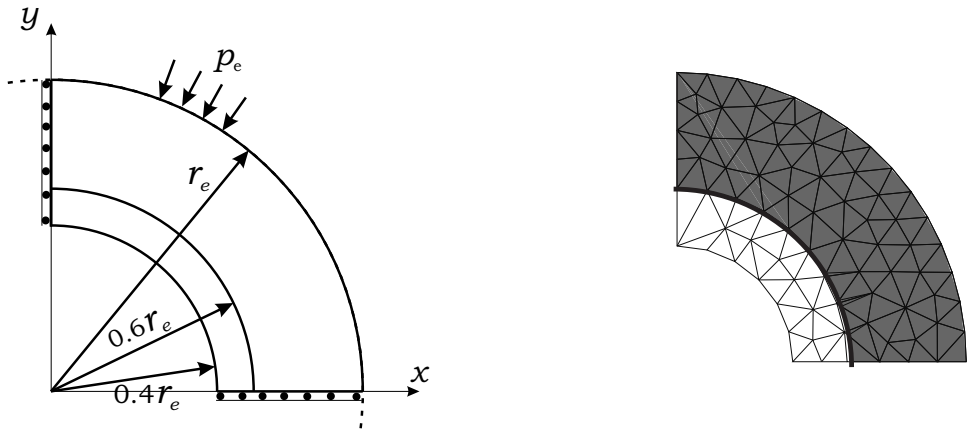
The sensitivity analysis implemented in OOFELIE performs quite well as the relative error between finite difference and semi-analytical approach is always smaller than 1%. We have to notice that, during the numerical experiments of Tables 5.1 and 5.2, no new element is created between the initial and the preturbed state. The same approach has been repeated with other values of the perturbation $\delta$, and the element creation phenomenon appeared only once for a perturbation of $\delta$=$5.10^{-2}$ of the angle $\theta$. In that case, two elements are created and the relative

error is equal to 6 %. In this case, we have to remark that the validity of the semi-analytical approach becomes quite questionnable. Actually, if we perform another comparison for a perturbation of $\delta$=4.10$^{-2}$, the relative error is now equal to 4.8 % without any creation of elements. Therefore we can conclude that even if elements are created, which rarely occurs, the error induced by the semi-analytical approach is limited if the number of created elements remains small.

An error is obviously introduced by the semi-analytical strategy because the contributions related to new created elements are ignored. However, in practice, the contribution of these elements remains so small that the neglected contribution to the stiffness matrix does not have generally a significant effect on the accuracy of the sensitivity. In the following chapter 6, we will see that the strategy to ignore the creation of the new DOFs is reasonable and that, in practice, when solving real life applications, we did not encountered any major problems.

## 5.7 Mitigating the error of the semi-analytical sensitivity approach

Despite the fact that no problem was encountered in applications, it is really preferable to have the guarantee of obtaining the best accuracy for the sensitivities. To mitigate this SA problem, we focus on two improvements to obtain a more accurate sensitivity analysis: 1/replacing the void material by a soft one and 2/adapting the perturbation step.




(a) Geometry and boundary conditions (b) Critical mesh

Figure 5.8: Hollow cylinder structure for the computation a compliance sensitivity

As we have shown in the previous section, resorting to semi-analytical approach can come into trouble when computing SA derivatives. To estimate the error that occurs when ignoring the creation of new elements, we consider the following test case. The structure is a hollow cylinder of external and internal radius $r_e$ and $0.4r_e$ under an external radial pressure $p$. By symmetry, we are able to study only one quarter of the structure by imposing suitable symmetry boundary conditions along $x$ and $y$ axis as represented in Figure 5.8.

In order to enforce a 'pathological' case, we deliberately introduce an inner boundary circle inside the hollow cylinder at $r$=$r_a$=$0.6r_e$ to force a conforming mesh along this boundary. Then, a circular Level Set with a radius $r_{ls} = r_a + \delta r/r_a$, $\delta r \ll r_a$, is introduced.

If we compute the sensitivity analysis with respect to the radius of the Level Set using a perturbation $\delta r<0$, it is likely that the perturbed design will introduce new DOFs. In Table 5.3, we compare the sensitivity of the compliance with a finite difference scheme and a semi-analytic using *ignore* strategy for a positive and a negative perturbation $\delta r$.

| Compliance sensitivity | | | | |
|---|---|---|---|---|
| *Method* | r | $\delta r$ | *Value* | *Relative error (%)* |
| *FD* | 0.6 | $10^{-4}r$ | 1.677E-5 | – |
| *FD* | 0.6 | $-10^{-4}r$ | 1.620E-5 | 3.3% |
| *SA* | 0.6 | $10^{-4}r$ | 1.676E-5 | 0.025% |
| *SA* | 0.6 | $-10^{-4}r$ | 2.794E-6 | 83% (16 new elements) |

Table 5.3: Semi-analytical sensitivity analysis approximation with *ignore* strategy. Relative error expressed with forward $FD$ as reference solution.

As expected the finite difference method gives similar results independently of the perturbation sign whereas the semi-analytic approach gives good agreement only when $\delta r$ is positive. However, when $\delta r$ is negative, new elements (16 new elements) and new DOFs appear in the model and the value of the sensitivity is highly under estimated.

The first conclusion that is clearly obvious is that the sign of perturbation is important when using the strategy in which we ignore the new DOFs. As a consequence, it is always safer to choose the perturbation step sign in order to compute sensitivity in a direction that moves into the structural domain and not out of the structure as pointed in this test case.

Unfortunately, modifying the sign of the perturbation can not ensure in all cases that no new DOFs are introduced by the perturbation of a design variable. For instance, if the design variable represents the position of a circle, any sign of the perturbation creates a modification that goes into the material on one side and outside of the material on the other side as illustrated in Figure 5.9 with the dashed zone.

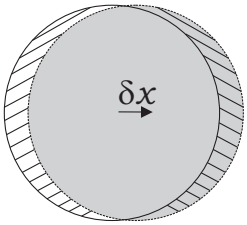



Figure 5.9: Creation of material when perturbing the position of the center of a circle

### 5.7.1 Introduction of soft material boundary layer

In topology optimization, the void is approximated by a material with a very small density. In this case, the mesh is also fixed but the size of the problem remains constant as the introduction of holes is realized with a mollifier function (similar to SIMP law for instance). Even if a huge zone with a very low density exists, all elements are kept in the formulation.

To circumvent the introduction of new DOFs, we can also introduce a soft material in place of the void. However, to limit the influence of the soft material, we can keep only a narrow band

(boundary layer) of elements with very soft mechanical properties around the Level Set $\psi = 0$ in order to prevent the variation of the total number of degrees of freedom. When applied to the test case of the hollow cylinder with a Young Modulus ratio of $10^{-3}$ between material zone and void zone, we obtain a relative difference on the strain energy of 0.07%.

The Table 5.4 indicates the values of the sensitivity obtained with this approximation, compared to the $FD$ results of Table 5.3 (e.g. with real void).

| Compliance sensitivity | | | | |
|---|---|---|---|---|
| *Method* | r | $\delta r$ | *Value* | *Relative error (%)* |
| *FD* | 0.6 | $10^{-4}r$ | 1.67165E-5 | 0.3% |
| *FD* | 0.6 | $-10^{-4}r$ | 1.69524E-5 | 1.08 % |
| *SA* | 0.6 | $10^{-4}r$ | 1.67124E-5 | 0.3% |
| *SA* | 0.6 | $-10^{-4}r$ | 1.68921E-5 | 0.7% |

Table 5.4: Comparison of sensitivity analysis with soft material in void zone and $FD$ forward scheme of Table 5.3 as reference solution.

This soft material strategy has the advantage of keeping the number of degrees of freedom constant and to forbid the creation of elements during the perturbation step. Hence, the computation of the sensitivity leads to a more accurate result as all elements are taken into account in the perturbed stiffness matrix as we can convince in Table 5.4. However, the presence of these elements is expected to introduce a dependency upon the mechanical properties associated to the narrow softening elements band like in topology optimization with the power $p$ coefficient in the $SIMP$ law. Particular difficulties could be encountered specifically in vibration problems where the phenomenon of local vibration modes could occur [136]. Furthermore, using these methods does not take fully advantage of the X-FEM as we re-introduce an approximation of the void as a soft material and so we loose the black/white description.

### 5.7.2 Adapting the perturbation

Imagine that the following Figure 5.10 represents the evolution of the stiffness matrix with respect to the design variable $z$. Moreover, consider that the stiffness matrix has a dimension $n$ and $n+1$ at the positions $z$ and $z+\delta z$ respectively. $z+\delta z^*$ is the value of the perturbation where the stiffness matrix changes its dimension. For instance, one can think of a 1D model composed of 2 elements where only the first element is filled with material over a length $z$ as represented in Figure 5.10 (b).

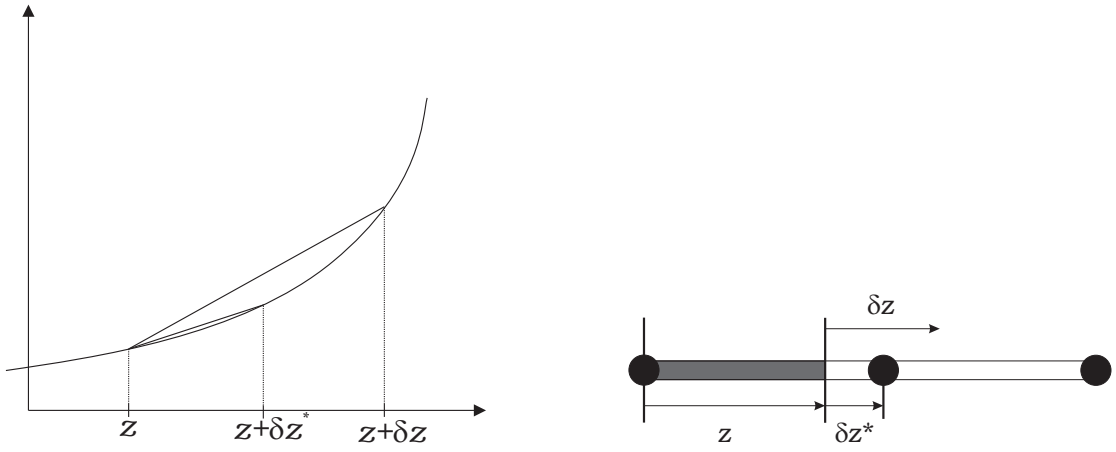
If we use the *ignore* strategy, the stiffness derivative computed is in fact equal to:

$$\frac{\partial K_{ignore}}{\partial z} \simeq \frac{\delta K_{ignore}}{\delta z} = \frac{K(z+\delta z^*) - K(z)}{\delta z} \tag{5.50}$$

as we do not take into account the contribution of the new elements that creates a new DOFs. Now writing the expression of the 'exact' stiffness derivative we have:

$$\frac{\partial K}{\partial z} \simeq \frac{\delta K}{\delta z} = \frac{K(z+\delta z) - K(z)}{\delta z} \tag{5.51}$$

introducing the zero term $K(z+\delta z^*)-K(z+\delta z^*)$ and rearranging the different terms into (5.51),




(a) Evolution of the stiffness matrix (b) 1D model

Figure 5.10: Sensitivity difficulty with semi-analytic approach

we obtain:

$$\begin{aligned}\frac{\partial K}{\partial z} &\simeq \frac{K(z+\delta z) - K(z+\delta z^*) + K(z+\delta z^*) - K(z)}{\delta z} \\ &\simeq \frac{K(z+\delta z^*) - K(z)}{\delta z} + \frac{K(z+\delta z) - K(z+\delta z^*)}{\delta z} \qquad (5.52) \\ &= \frac{\partial K_1}{\partial z} + \frac{\partial K_2}{\partial z} \qquad (5.53)\end{aligned}$$

From expression (5.52) we can observe that the exact stiffness derivative can be divided into the sum of two parts $\frac{\partial K_1}{\partial z}$ and $\frac{\partial K_2}{\partial z}$ corresponding to the stiffness derivative of the first and second element respectively. Moreover, comparing (5.52) and (5.50) we can see that the ignore strategy corresponds to the stiffness derivative on the first element and that the error of this approach is equal to the contribution of the second element. Furthermore, we can see that this error increases with the distance $|\delta z - \delta z^*|$ and is null if $\delta z = \delta z^*$. Hence, if we correct the perturbation step $\delta z$ to be equal to $\delta z^*$ in (5.52) or replace $\delta z$ by $\delta z^*$ in (5.50), the sensitivity analysis can be computed accurately if the perturbation $\delta z^*$ is not too small compared to computer precision (see [81]).

In [183], Zhang indicates that it is more effective to compute the derivative by assembling first the difference of matrices $K(z+\delta z) - K(z)$ at element level and then by dividing the result by the perturbation $\delta z$. The perturbation has therefore to be unique for all elements. However, we can also compute a sensitivity element by element and then assemble the elementary stiffness derivatives allowing therefore to prescribe a perturbation for each elements.

Practically, it is not necessary to compute an adapted perturbation on each element as most of them keep the same state. Hence, in the *ignore* strategy (in which we ignore the new DOFs), we have to compare the state of the elements (e.g. cut, solid) at $z$ and $z+\delta z$ and take into account only the elements that keep a constant or decreasing number of DOFs. With a sensitivity analysis computed on each element, we can focus solely on the elements that are cut at the state $z$. Then, given the perturbation $\delta z$, we compute the sensitivity on these cut elements only. When this perturbation $\delta z$ modifies the state of the element such that the iso-zero Level Set goes out of the element, an adapted perturbation $\delta z^*$ should be computed. Here again, the Level Set description

greatly helps the detection of the state modification of elements as depicted in Figure 5.11. A simple product of nodal Level Set Values $\phi_i(z)$ and $\phi_i(z+\delta z)$ can inform on the modification of the topology and the state of the element.

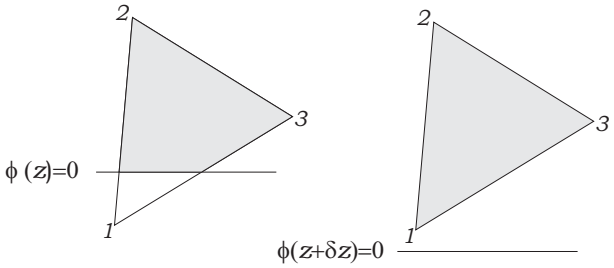



Figure 5.11: Modification of the state of the element

Given the state $z$ and $z+\delta z$, we can compute the derivative of the Level Set $\phi$ for each node $i$ using:

$$\frac{\partial \phi_i}{\partial z} \simeq \frac{\delta \phi_i}{\delta z} = \frac{\phi_i(z+\delta z) - \phi_i(z)}{\delta z}$$

The perturbation $\delta z$ is supposed to be infinitesimal, hence, from the nodal Level Set values $\phi_i$ we can compute the maximum perturbation $\delta z^{max}$ that preserves the same cut state (elementary topology) for the element:

$$\phi(z+\delta z^{max}) = 0 = \phi(z) + \frac{\partial \phi}{\partial z}\delta z^{max} \Leftrightarrow \delta z^{max} = -\left(\frac{\partial \phi}{\partial z}\right)^{-1}\phi(z)$$

Practically, the $\delta z^{max}$ is computed by considering the closest negative node from the boundary:

$$\delta z^{max} = \frac{\delta z}{\delta \phi} * \frac{\max(\phi_i(z) < 0)}{\alpha}; \tag{5.54}$$

where $\max(\phi_i(z) < 0)$ means the maximum negative nodal Level Set value and $\alpha$ is a factor $> 1$ to ensure that the perturbation step $\delta z^{max}$ is not too large. Given this $\delta z^{max}$, we can regenerate new Level Set values using the value of the parameter $\delta z^{max}$ or, more efficiently, approximate the new Level Set values on the element using:

$$\phi_i(z+\delta z^{max}) = \phi_i(z) + \frac{\partial \phi_i}{\partial z}\delta z^{max}$$

Applying this method to the hollow cylinder test case with an initial perturbation of $\delta r = -10^{-4}$, we can see in the Table 5.5 that we recover a sensitivity that is in perfect agreement with the finite difference method.

| Compliance sensitivity | | | | |
|---|---|---|---|---|
| *Method* | r | *δr* | *Value* | *Relative error (%)* |
| *FD* | 0.6 | $10^{-4}r$ | 1.677E-5 | – |
| *SA* | 0.6 | $-10^{-4}r$ | 2.794E-6 | 83 % |
| *Semi-analytic adapted* | $r = 0.6$ | $10^{-4}r$ | 1.677E-5 | 0.016% |

Table 5.5: Validation of the semi-analytical adapted sensitivity analysis approximation.

One inherent drawback of this method is that there is no guarantee that the adapted perturbation step $\delta z^{max}$ is not close to computer precision leading to an erroneous elementary structural matrices derivative. The procedure to adapt the perturbation can be summarized as the following algorithm:

**Algorithm 1** : Computation of sensitivity analysis using adapted perturbation step

```
Generate the Level Set, and cut the mesh for configuration z and z + δz.
for all Cut elements do
    Compute state for configuration z and z + δz
    if The state does not change then
        Compute the sensitivity with finite difference on elementary matrices between configuration z and z + δz.
    else
        Compute adapted perturbation step δz^max using (5.54)
        Compare state of elements at z and z + δz^max
        Approximate perturbed Level Set values at z + δz^max or regenerate a Level Set only on the element
        Compute the sensitivity with finite difference on elementary matrices between configuration z and z + δz^max
    end if
    Assemble the elementary derivatives
end for
Compute objective function or constraint sensitivity given the global structural derivatives
```

## 5.8 Analytical derivatives

Analytically, the expression of the stiffness matrix of an extended finite element can be written exactly as a classical FE. Hence, the derivation of the stiffness matrix can be obtained following the same procedure as for FEM (see section 5.3.3). However, due to the fixed grid approach, the effect of the shape parameter '$z$' is different as it only involves a modification of the domain of integration $\Omega_{solid}(z)$ and does not introduce any modification of the elementary geometry. Furthermore, when the approximation of the displacement field involves additional shape functions such as the ones used for crack and bi-material enrichment, new terms, depending on the parameter '$z$', have to be taken into account.

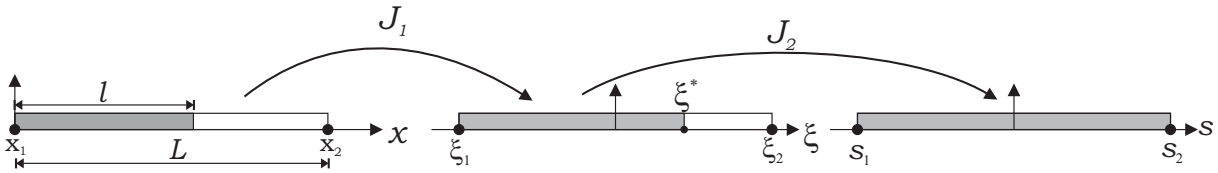



Figure 5.12: Linear rod element mapping in X-FEM

Hence, given the specific integration procedure of X-FEM, the expression of the stiffness matrix can take three different forms depending on the space in which it is written. Expressed in the

real space, or in the common referential space, one has two similar forms:

$$K = \int_{\Omega_{solid}(z)} B^T H B d\Omega_{solid} \tag{5.55}$$

and

$$K = \int_{\xi^*(z)} B^T H B \left|J_1\right| d\xi \tag{5.56}$$

or also

$$K = \int_{s_1}^{s_2} B^T H B \left|J_1\right| \left|J_2\right| ds \tag{5.57}$$

where both domains of integration $\Omega_{solid}(z)$ in referential $x$ and $\xi^*$ in referential $\xi$ depend on the parameter '$z$'. Consequently, a material derivative approach should be used in order to obtain the sensitivity of $K$ from these two expressions. One can clearly notice here a first difference between FEM and X-FEM as the introduction of the common reference space does not yield to a fixed integration space in the referential $\xi$. Applying the material derivative identity (5.22) to the stiffness matrix, one has:

$$\frac{dK}{dz} = \int_{\Omega(z)} \frac{d}{dz}(B^T H B)\ d\Omega + \int_{\Omega(z)} B^T H B div V\ d\Omega \tag{5.58}$$

In our case, it is more useful to rewrite this expression by splitting the total derivative $\frac{d}{dz}\cdot$ into $\frac{\partial}{\partial z}\cdot + V\nabla\cdot$:

$$\frac{dK}{dz} = \int_{\Omega(z)} \frac{\partial}{\partial z}(B^T H B)\ d\Omega + \int_{\Omega(z)} div(B^T H B V)\ d\Omega \tag{5.59}$$

or using the Gauss theorem:

$$\frac{dK}{dz} = \int_{\Omega(z)} \frac{\partial}{\partial z}(B^T H B)\ d\Omega + \int_{\Gamma(z)} B^T H B V n\ d\Gamma \tag{5.60}$$

However, we find that because there is no explicit dependence of the shape functions on the shape parameter, the partial derivative $\frac{\partial}{\partial z}\cdot$ is equal to zero.

$$\frac{d}{dz}(\frac{\partial}{\partial x_i}N_j) = \frac{\partial}{\partial z}(\frac{\partial}{\partial x_i}N_j) + V_k\frac{\partial}{\partial x_k}(\frac{\partial}{\partial x_i}N_j) = V_k\frac{\partial}{\partial x_k}(\frac{\partial}{\partial x_i}N_j) \tag{5.61}$$

Therefore, in (5.58) and (5.59), the first term is zero and only convective terms have to be taken into account. So, the stiffness matrix derivative reduces to:

$$\begin{aligned}\frac{dK}{dz} &= \int_{\Omega_{solid}(z)} div(B^T H B V) d\Omega_{solid} \\ &= \int_{\Gamma(z)} B^T H B V n\ d\Gamma\end{aligned} \tag{5.62}$$

Note that this is true only in the case of void-material approximation for X-FEM as detailed in section 5.11.

Reintroducing the example of a 1D rod element partially filled along a length $l$, the stiffness matrix is defined as :

$$K = \int_0^l \begin{bmatrix} -\frac{1}{L} \\ \frac{1}{L} \end{bmatrix} EA \begin{bmatrix} -\frac{1}{L} & \frac{1}{L} \end{bmatrix} dx = \frac{EAl}{L^2}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

and one can clearly see that all the terms under the integral term do not depend on the shape design variable $l$. The derivation of this expression with respect to the parameter $z{=}l$ gives:

$$\frac{\partial K}{\partial l} = \frac{EA}{L^2}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \tag{5.63}$$

To apply equation (5.62), one should define the velocity field $V$ describing the movement of a material point for a design variable representing the length of the material zone $l$. This one can be chosen as:

$$V = \frac{x}{l}$$

and introduced into (5.62) to obtain an expression equivalent to analytical expression (5.63):

$$\begin{aligned}
\frac{\partial K}{\partial l} &= EA\{\int_0^l \frac{\partial B}{\partial x}^T BV(x)dx + \int_0^l B^T \frac{\partial B}{\partial x} V(x)dx + \int_0^l B^T B\, divV(x)dx\} \\
&= EA\int_0^l \left\{ \begin{bmatrix} 0 \\ 0 \end{bmatrix} \begin{bmatrix} -\frac{1}{L} & \frac{1}{L} \end{bmatrix} \frac{x}{l} + \begin{bmatrix} -\frac{1}{L} \\ \frac{1}{L} \end{bmatrix} \begin{bmatrix} 0 & 0 \end{bmatrix} \frac{x}{l} + \begin{bmatrix} -\frac{1}{L} \\ \frac{1}{L} \end{bmatrix} \begin{bmatrix} -\frac{1}{L} & \frac{1}{L} \end{bmatrix} \frac{1}{l} \right\} dx \\
&= \frac{EA}{L^2}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}
\end{aligned}$$

Written in the standard reference $\xi$ space, the situation is the same since the domain of integration is also a function of the shape parameters and that the introduction of the first Jacobian $J_1$, based only on the geometry of the full element has no direct relation with the domain of integration.

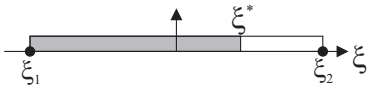



Figure 5.13: Rod element in the $\xi$ referential

The integral has the same form as in real space and the bounds of integration also depend on the shape parameter. Obviously here the bound of integration $\xi^* = \frac{2l}{L} - 1$ depends on the parameter $l$:

$$\begin{aligned}
K &= \int_{\xi_1}^{\xi^*} (J_1^{-1})^T \begin{bmatrix} \frac{\partial N_1(\xi)}{\partial \xi} \\ \frac{\partial N_2(\xi)}{\partial \xi} \end{bmatrix} E \begin{bmatrix} \frac{\partial N_1(\xi)}{\partial \xi} & \frac{\partial N_2(\xi)}{\partial \xi} \end{bmatrix} J_1^{-1} |J_1|\, d\xi \\
&= \int_{-1}^{\frac{2l}{L}-1} \frac{2}{L} \begin{bmatrix} -\frac{1}{2} \\ \frac{1}{2} \end{bmatrix} EA \begin{bmatrix} -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \frac{2}{L} \frac{L}{2} d\xi
\end{aligned}$$

Hence, conversely to a FEM approach, the introduction of the first mapping does not yield to a fixed integration domain. As mentioned in chapter 4, these two formulations are not easy tractable due to the fact that the bounds of integration are not fixed and the classical integration scheme can not be used as it is. To overcome this problem, a second mapping is introduced to obtain the practical formulation used within X-FEM codes:

$$K = \int_s B^T H B\, |J_1|\, |J_2|\; ds$$

Introducing the second mapping has the same consequence as the introduction of the unique mapping used with FEM. The domain of integration remains constant with respect to a modification

of the design variable but the terms under the integral are now depending on this parameter. One obtains thus a formulation where the integration domain is independent of $l$ but not the term inside this integral as both $J_2$ and $B$ matrix depend on $l$ through the mapping:

$$\begin{aligned}
\xi &= \frac{1}{2}(1-s)\xi_1 + \frac{1}{2}(1+s)\xi^* \\
&= -\frac{1}{2}(1-s) + \frac{1}{2}(1+s)(\frac{2l}{L} - 1) \\
K &= \int_{-1}^{1} J_1^{-1} \begin{bmatrix} \frac{\partial N_1(\xi(l))}{\partial \xi} \\ \frac{\partial N_2(\xi(l))}{\partial \xi} \end{bmatrix} EA \begin{bmatrix} \frac{\partial N_1(\xi(l))}{\partial \xi} & \frac{\partial N_2(\xi(l))}{\partial \xi} \end{bmatrix} J_1^{-1} \left|J_1\right| \left|J_2\right| \, ds \\
&= \int_{-1}^{1} \frac{2}{L} \begin{bmatrix} -\frac{1}{2} \\ \frac{1}{2} \end{bmatrix} EA \begin{bmatrix} -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \frac{2}{L} \frac{L}{2} \frac{l}{L} \, ds
\end{aligned}$$

To generalize the formulation of the stiffness matrix derivative and to obtain a practical formulation for the stiffness matrix derivative, let's focus on the discrete formulation, which is computed using the following form:

$$\begin{aligned}
K = \int_{\Omega_{Solid}} B^T H B d\Omega_{Solid} &= \sum_{\Delta_i} \int_{\Delta_i} B^T H B \left|J_1\right| \left|J_2\right| d\xi d\eta \\
&= \sum_{\Delta_i} \sum_{gpt}^{n_{gpt}} B^T H B \left|J_1\right| \left|J_2\right| w_{gpt}
\end{aligned}$$

The derivation with respect to a shape parameter '$z$' can be computed under the integration sign since it is carried out over a fixed domain:

$$\begin{aligned}
\frac{dK}{dz} &= \frac{\partial}{\partial z} \sum_{\Delta_i} \sum_{gpt}^{n_{gpt}} B^T H B \left|J_1\right| \left|J_2\right| w_{gpt} \\
&= \sum_{\Delta_i} \sum_{gpt}^{n_{gpt}} w_{gpt} \left\{ \frac{\partial B^T}{\partial z} H B \left|J_1\right| \left|J_2\right| + B^T H \frac{\partial B}{\partial z} \left|J_1\right| \left|J_2\right| + B^T H B \left|J_1\right| \frac{\partial \left|J_2\right|}{\partial z} \right\}
\end{aligned}$$

The $B$ matrix for the node $r$ has the following expression:

$$B_r = \begin{bmatrix} \frac{\partial N_{r,x}}{\partial z} \\ \frac{\partial N_{r,y}}{\partial z} \end{bmatrix} = J_1^{-1} \begin{bmatrix} N_{r,\xi} \\ N_{r,\eta} \end{bmatrix}$$

and the derivative of $B$ can be easily obtained using the relation $d(X^{-1}) = -X^{-1} dX X^{-1}$ [61]:

$$\frac{\partial B_r}{\partial z} = \begin{bmatrix} \frac{\partial N_{r,x}}{\partial z} \\ \frac{\partial N_{r,y}}{\partial z} \end{bmatrix} = -J_1^{-1} \frac{\partial J_1}{\partial z} J_1^{-1} \begin{bmatrix} N_{r,\xi} \\ N_{r,\eta} \end{bmatrix} + J_1^{-1} \begin{bmatrix} \frac{\partial N_{r,\xi}}{\partial z} \\ \frac{\partial N_{r,\eta}}{\partial z} \end{bmatrix} = J_1^{-1} \begin{bmatrix} \frac{\partial N_{r,\xi}}{\partial z} \\ \frac{\partial N_{r,\eta}}{\partial z} \end{bmatrix}$$

The first term is null in this case because the Jacobian $J_1$ does not change with a perturbation of $z$ conversely to a shape parameter with FEM where the geometry of the element changes. Using the chain rule derivative, we can express the derivative $\partial N_\xi / \partial z$ to obtain the following expression:

$$\frac{\partial B_r}{\partial z} = J_1^{-1} \begin{bmatrix} \frac{\partial N_{r,\xi}}{\partial \xi} \frac{\partial \xi}{\partial z} + \frac{\partial N_{r,\xi}}{\partial \eta} \frac{\partial \eta}{\partial z} \\ \frac{\partial N_{r,\eta}}{\partial \eta} \frac{\partial \eta}{\partial z} + \frac{\partial N_{r,\eta}}{\partial \xi} \frac{\partial \xi}{\partial z} \end{bmatrix}$$

And one can notice that, in this expression, the second derivative of the shape functions $\partial N_\xi/\partial\xi$ and a velocity field term in the reference element $\partial\xi/\partial z$ appears.

Identically to shape parameters with FEM, the last term of (5.64) is the contribution of the modification of the integration domain, which is now relative to the derivative of the determinant of the second Jacobian.

$$\frac{\partial \det(J_2)}{\partial z} = \det \frac{\partial}{\partial z} \begin{vmatrix} \sum_i^n N_{i,s}\xi_i & \sum_i^n N_{i,s}\eta_i \\ \sum_i^n N_{i,t}\xi_i & \sum_i^n N_{i,t}\eta_i \end{vmatrix}$$

or, explicitly:

$$\begin{aligned} \frac{\partial \det(J_2)}{\partial z} = &\sum_i^n N_{i,s}V_{i\xi} \sum_i^n N_{i,t}\eta_i + \sum_i^n N_{i,s}\xi_i \sum_i^n N_{i,t}V_{i\eta} \\ &- \sum_i^n N_{i,t}V_{i\xi} \sum_i^n N_{i,s}\eta_i - \sum_i^n N_{i,t}\xi_i \sum_i^n N_{i,s}V_{i\eta} \end{aligned}$$

where $V_{i\xi}$ is the velocity field of node $i$ along the $\xi$ direction.

Hence, compared to a classical shape parameter problem, the sensitivity of the stiffness matrix introduces a velocity field that can be expressed in the reference space $(\xi, \eta)$ corresponding to the movement of a material point. Moreover, the sensitivity of the Jacobian is now on the second Jacobian $J_2$ where a velocity field also appears.

Numerically, one can see that the modification of a parameter '$z$' has the effect of changing the integration domain and also the position of the sampling points. Indeed, the second derivative of the shape functions $\partial N_\xi/\partial\xi$ represent the variation of the derivative of the shape functions when the point of evaluation is modified. The second term $\partial\xi/\partial z$ represents the variation of the Gauss point position that is introduced with the variation of the parameter '$z$' and can be interpreted as a velocity field. Finally, the term including the derivative of the second Jacobian takes into account of the integral domain modification.

## 5.9 Numerical applications

### 5.9.1 Linear elements

When the element is based on linear shape functions like in rods, triangles and tetrahedron, the XFEM stiffness matrix can be obtained by a scaling with respect to the Finite Element stiffness matrix. Indeed, in this case, the term $B^T H B$ is constant (as well as the Jacobian) and it can be taken out of the integral to give:

$$K_{FEM} = B^T H B \int_\Omega d\Omega = B^T H B V_{element}$$

where $K_{FEM}$ is the stiffness matrix of a classic finite element and $V_{element}$ the volume of the element. Given the expression of the stiffness matrix of the extended element, one has:

$$K = B^T H B \left( \int_{\Omega_{solid}} d\Omega \right) = B^T H B V_{solid} = K_{FEM} \frac{V_{solid}}{V_{element}}$$

where $V_{Solid}$ the volume of the filled part. This expression is very useful as it indicates that the stiffness matrix can be computed very quickly. One solely needs to the expression of stiffness matrix $K_{FEM}$ of the solid element and to compute the volume of the element and the filled part. Moreover, we can obtain directly the sensitivity of the stiffness matrix by computing the sensitivity of the volume:

$$\frac{dK}{dz} = \frac{K_{FEM}}{V_{element}} \frac{dV_{solid}}{dz} \tag{5.64}$$

Also, if one has the expression of the velocity field, (5.64) can be computed analytically with a domain or a boundary integral using the analytical derivative of the volume

$$\frac{dV}{dz} = \int_{\Omega_{solid}} divV \, d\Omega = \int_{\Gamma_{solid}} V.n \, d\Gamma_{solid}$$

However, please notice that these results are only valid for the stiffness matrix and for approximations that are strictly linear. For instance, bi-linear elements can not be integrated using this approach as the term $B^T H B$ is not constant anymore due to the '$xy$' component of the shape functions.

### 5.9.2 Quadratic 1D element

With second degree shape functions, the terms $\frac{\partial B_r}{\partial z}$ are not null anymore and the stiffness matrix derivative is now composed of three terms. To illustrate this, let's consider a 1D second order element, the shape function are defined as:

$$\begin{aligned} N_1(\xi) &= -\frac{1}{2}\xi(1-\xi) \\ N_2(\xi) &= \frac{1}{2}\xi(1+\xi) \\ N_3(\xi) &= (1-\xi^2) \end{aligned}$$

And the derivatives are:

$$\begin{array}{ll} \frac{\partial N_1}{\partial \xi} = -\frac{1}{2}(1-2\xi) & \frac{\partial}{\partial \xi}\frac{\partial N_1}{\partial \xi} = 1 \\ \frac{\partial N_2}{\partial \xi} = \frac{1}{2}(1+2\xi) & \frac{\partial}{\partial \xi}\frac{\partial N_2}{\partial \xi} = 1 \\ \frac{\partial N_3}{\partial \xi} = -2\xi & \frac{\partial}{\partial \xi}\frac{\partial N_3}{\partial \xi} = -2 \end{array}$$

The analytical expression of the stiffness matrix becomes:

$$\begin{bmatrix} \frac{27\mathrm{L}^2 l - 26Ll^2 + 16l^3}{3L^4} & \frac{3Ll - 4l^2}{3L^4} & \frac{-36\mathrm{L}^2 l - 60Ll^2 - 32l^3}{3L^4} \\ \frac{3Ll - 4l^2}{3L^4} & \frac{3\mathrm{L}^2 l - 12Ll^2 + 16l^3}{3L^4} & \frac{-12\mathrm{L}^2 l - 36Ll^2 - 32l^3}{3L^4} \\ \frac{-36\mathrm{L}^2 l - 60Ll^2 - 32l^3}{3L^4} & \frac{-12\mathrm{L}^2 l - 36Ll^2 - 32l^3}{3L^4} & \frac{-54\mathrm{L}^2 l - 96Ll^2 - 64l^3}{3L^4} \end{bmatrix}$$

The stiffness derivative in the reference space can be expressed as ($z$=$l$):

$$\frac{\partial K}{\partial l} = EA\int_{-1}^{\xi^*}\left(\frac{\partial B}{\partial \xi}\frac{\partial \xi}{\partial l}\right)^T B\,|J_1|\,|J_2|\,d\xi + \int_{-1}^{\xi^*} B^T\frac{\partial B}{\partial \xi}\frac{\partial \xi}{\partial l}\,|J_1|\,|J_2|\,d\xi \tag{5.65}$$

$$+EA\int_{-1}^{\xi^*} B^T B div V(\xi)\,|J_1|\,d\xi$$

$$= EA\int_{-1}^{\xi^*}\left(\frac{\partial B}{\partial \xi}V(\xi)\right)^T B\,|J_1|\,|J_2|\,d\xi + EA\int_{-1}^{\xi^*} B^T\frac{\partial B}{\partial \xi}V(\xi)\,|J_1|\,|J_2|\,d\xi \tag{5.66}$$

$$+EA\int_{-1}^{\xi^*} B^T B div V(\xi)\,|J_1|\,d\xi$$

$$= EA\int_{-1}^{\xi^*}\{(J_1^{-1})^T\begin{bmatrix}1\\1\\-2\end{bmatrix}V(\xi)J_1^{-1}\begin{bmatrix}-\frac{1}{2}\xi(1-\xi) & \frac{1}{2}\xi(1+\xi) & (1-\xi^2)\end{bmatrix}|J_1|\,|J_2|$$

$$+(J_1^{-1})^T\begin{bmatrix}-\frac{1}{2}\xi(1-\xi)\\ \frac{1}{2}\xi(1+\xi)\\ (1-\xi^2)\end{bmatrix}J_1^{-1}\,[1\;1\;-2]\,V(\xi)\,|J_1|\,|J_2|$$

$$+(J_1^{-1})^T\begin{bmatrix}-\frac{1}{2}\xi(1-\xi)\\ \frac{1}{2}\xi(1+\xi)\\ (1-\xi^2)\end{bmatrix}J_1^{-1}\left[-\frac{1}{2}\xi(1-\xi)\;\frac{1}{2}\xi(1+\xi)\;(1-\xi^2)\right]|J_1|\,div V(\xi)\}d\xi \tag{5.67}$$

where $\xi^*$ is the position of the void solid interface inside the reference element in the $\xi$ referential. The velocity field can be chosen as before:

$$V(x) = \frac{x}{l}$$

However, in our numerical implementation, the high order elements are first divided into sub-element as explained in chapter 4. Hence, the 1D second degree element is first divided into two integration elements based on the nodes 0-1 and 1-2 (see Fig. 5.14). Given a modification of the position of the cut position '$l$', one can remark (see Fig. 5.14) that the integration element 0-1 is not affected by a perturbation $\delta l$, while the second integration 1-2 is modified. From this observation, one can define another discontinuous velocity field as:

$$V(x) = \left\{\begin{array}{ll} 0 \quad \text{if} & x < \frac{L}{2} \\ \frac{2x-L}{2l-L} \quad \text{if} & \frac{L}{2} \le x \le l \end{array}\right\}$$

which is schematically represented in Figure 5.14.

In the referential $\xi$, we have:

$$V(\xi) = \frac{\xi}{2l}$$

It is even clearer if we express the velocity field in the referential $s$. For the second integration sub-element:

$$V(s) = V_1(1-s) + V_2 s$$

with $V_1$ the velocity field at $s = -1$, or $\xi = 0$ and we have $V_1 = 0$. $V_2$ is the velocity field at $s = 1$ or $\xi = 2l$ and $V_2 = 1$.

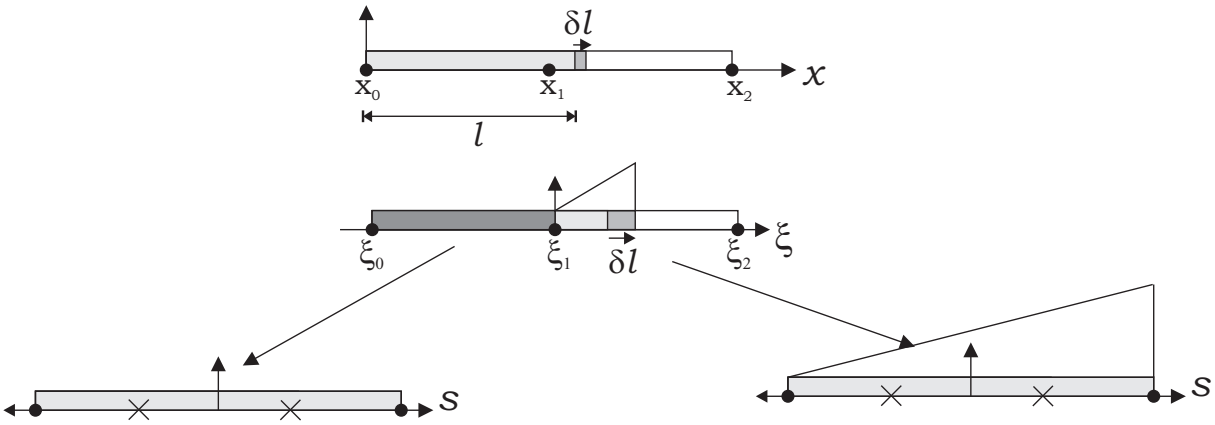



Figure 5.14: Perturbation of $l$ on the reference line space

The expression (5.67) can then be completed, limiting the integration between 0 and $\xi^*$ as the velocity field is zero between -1 and 0. The Jacobian $J_2$ on the second integration element $0-\xi^*$ is equal to 0.1 because the distance $0-\xi^*$ is equal to $2(L-l)=0.2$. The Jacobian $J_1=\frac{dx}{d\xi}$ is equal to 0.5 as $L=1$. Given a two Gauss points integration scheme defined as:

$$\begin{aligned} s^1 = &\quad -0.57745 \; w^1 = 1 \\ s^2 = &\quad 0.57745 \; w^2 = 1 \end{aligned}$$

in the referential '$s$'. We can have the value of the velocity field at integration points:

$$\begin{aligned} V(s^1) &= (1-s^1)V_1 + s^1 V_2 = s^1 V_2 \\ V(s^2) &= (1-s^2)V_1 + s^2 V_2 = s^2 V_2 \end{aligned}$$

In the $\xi$ referential, these points are mapped to:

$$\begin{aligned} \xi^1 = &\quad (1-s^1)\xi_1 + s^1\xi_2 = s^1\xi_2 = 0.042290 \\ \xi^2 = &\quad (1-s^2)\xi_1 + s^2\xi_2 = s^2\xi_2 = 0.157829 \end{aligned}$$

and the velocity field remains the same:

$$\begin{aligned} V(\xi^1) &= V(s^1) \\ V(\xi^2) &= V(s^2) \end{aligned}$$

Given all these terms, we can formulate the derivative of the stiffness matrix:

$$\begin{aligned} \frac{\partial K}{\partial l} &= EA \int_0^{\xi^*} [J_1^{-1} \begin{bmatrix} 1 \\ 1 \\ -2 \end{bmatrix} V(\xi) J_1^{-1} \left[ -\frac{1}{2}\xi(1-\xi) \;\; \frac{1}{2}\xi(1+\xi) \;\; (1-\xi^2) \right] |J_1| \, |J_2| \\ &+ J_1^{-1} \begin{bmatrix} -\frac{1}{2}\xi(1-\xi) \\ \frac{1}{2}\xi(1+\xi) \\ (1-\xi^2) \end{bmatrix} J_1^{-1} [1 \;\; 1 \;\; -2] \, V(\xi) \, |J_1| \, |J_2| \\ &+ J_1^{-1} \begin{bmatrix} -\frac{1}{2}\xi(1-\xi) \\ \frac{1}{2}\xi(1+\xi) \\ (1-\xi^2) \end{bmatrix} J_1^{-1} \left[ -\frac{1}{2}\xi(1-\xi) \;\; \frac{1}{2}\xi(1+\xi) \;\; (1-\xi^2) \right] |J_1| \, div V(\xi)] d\xi \end{aligned}$$

that yields:

```
dk/dl=3.60000000001083E-1   -8.40000000000301E-1    4.79999999999218E-1
     -8.40000000000301E-1    1.95999999999768      -1.11999999999738e
      4.79999999999218E-1   -1.11999999999738       6.39999999998167E-1
```

Using a finite difference method we have:

```
dk/dl=3.59856019197894E-1   -8.39903980799870E-1    4.80047961596656E-1
     -8.39903980799870E-1    1.96033601920035      -1.12043203840025
      4.80047961596656E-1   -1.12043203840025       6.40384076796192E-1
```

and one can remark that the two methods are in perfect agreement. The relative difference of the two solutions is:

```
rel_difference=
     3.99946675523805E-4    1.14308571942295E-4   -9.99199946620287E-5
     1.14308571942295E-4   -1.71438368706333E-4   -3.85748573986603E-4
    -9.99199946620287E-5   -3.85748573986603E-4   -6.00119996916633E-4
```

and the Frobenius norm ($\|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2}$) of the relative difference is 9.45E-4.

### 5.9.3 Quadrangle element

Consider now the example of a quadrangle that is cut along one reference direction. This element has bi-linear shape functions defined as:

$$
\begin{aligned}
N_1(\xi,\eta) &= \frac{1}{4}(1-\xi)(1-\eta)\\
N_2(\xi,\eta) &= \frac{1}{4}(1+\xi)(1-\eta)\\
N_3(\xi,\eta) &= \frac{1}{4}(1+\xi)(1+\eta)\\
N_4(\xi,\eta) &= \frac{1}{4}(1-\xi)(1+\eta)
\end{aligned}
$$

Due to the bi-linear properties of the shape functions, there are non zero second order derivatives of the shape functions:

$$
\begin{array}{llll}
\frac{\partial N_1}{\partial \xi} = \frac{1}{4}(\eta-1) & \frac{\partial}{\partial \eta}\frac{\partial N_1}{\partial \xi} = \frac{1}{4} & \frac{\partial N_1}{\partial \eta} = \frac{1}{4}(\xi-1) & \frac{\partial}{\partial \xi}\frac{\partial N_1}{\partial \eta} = \frac{1}{4}\\
\frac{\partial N_2}{\partial \xi} = \frac{1}{4}(1-\eta) & \frac{\partial}{\partial \eta}\frac{\partial N_2}{\partial \xi} = -\frac{1}{4} & \frac{\partial N_2}{\partial \eta} = -\frac{1}{4}(1+\xi) & \frac{\partial}{\partial \xi}\frac{\partial N_2}{\partial \eta} = -\frac{1}{4}\\
\frac{\partial N_3}{\partial \xi} = \frac{1}{4}(1+\eta) & \frac{\partial}{\partial \eta}\frac{\partial N_3}{\partial \xi} = \frac{1}{4} & \frac{\partial N_3}{\partial \eta} = \frac{1}{4}(1+\xi) & \frac{\partial}{\partial \xi}\frac{\partial N_3}{\partial \eta} = \frac{1}{4}\\
\frac{\partial N_4}{\partial \xi} = -\frac{1}{4}(1+\eta) & \frac{\partial}{\partial \eta}\frac{\partial N_4}{\partial \xi} = -\frac{1}{4} & \frac{\partial N_4}{\partial \eta} = \frac{1}{4}(1-\xi) & \frac{\partial}{\partial \xi}\frac{\partial N_4}{\partial \eta} = -\frac{1}{4}
\end{array}
$$

#### Horizontal cut

Perturbing the position of the cut in the element along the axis $y$ introduces a one-directional velocity field along this direction (see Fig. 5.15). Therefore, the contribution along the axis $x$ is zero and the velocity field can be chosen as:

$$V = \frac{y}{l}, \quad divV = \frac{1}{l}$$

Computing the different terms appearing in the stiffness derivative:

$$\frac{\partial B_r}{\partial l} = \frac{\partial B_r}{\partial \xi}\frac{\partial \xi}{\partial l} + \frac{\partial B_r}{\partial \eta}\frac{\partial \eta}{\partial l} = \frac{\partial B_r}{\partial \eta}\frac{\partial \eta}{\partial l}$$

because $\frac{\partial \xi}{\partial l} = 0$. For the node 1, we have:

$$\frac{\partial B_1}{\partial \eta} = \begin{bmatrix} \frac{\partial}{\partial \eta}\frac{\partial N_1}{\partial \xi} & 0 \\ 0 & \frac{\partial}{\partial \eta}\frac{\partial N_1}{\partial \eta} \\ \frac{\partial}{\partial \eta}\frac{\partial N_1}{\partial \eta} & \frac{\partial}{\partial \eta}\frac{\partial N_1}{\partial \xi} \end{bmatrix} = \begin{bmatrix} \frac{1}{4} & 0 \\ 0 & 0 \\ 0 & \frac{1}{4} \end{bmatrix}$$

Introducing these terms into (5.64), using a four Gauss points integration scheme and a subdivision based on a quadrangular integration element, we obtain:

```
dk/dl=
      0,46474  0,24038 -0,27244  0,04808 -0,40064 -0,24038  0,20833 -0,04808
      0,24038  0,54487 -0,04808  0,12821 -0,24038 -0,32051  0,04808 -0,35256
     -0,27244 -0,04808  0,46474 -0,24038  0,20833  0,04808 -0,40064  0,24038
      0,04808  0,12821 -0,24038  0,54487 -0,04808 -0,35256  0,24038 -0,32051
     -0,40064 -0,24038  0,20833 -0,04808  0,46474  0,24038 -0,27244  0,04808
     -0,24038 -0,32051  0,04808 -0,35256  0,24038  0,54487 -0,04808  0,12821
      0,20833  0,04808 -0,40064  0,24038 -0,27244 -0,04808  0,46474 -0,24038
     -0,04808 -0,35256  0,24038 -0,32051  0,04808  0,12821 -0,24038  0,54487
```

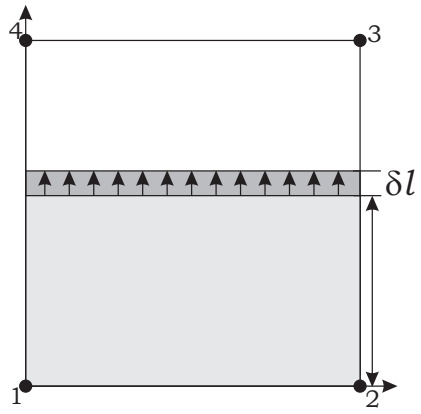



Figure 5.15: Perturbation of $l$ on the reference quadrangle

Using a difference method with a perturbation $\delta l$=$10^{-3}l$, we retrieve a perfect agreement as the Frobenius norm of the relative difference being 1.35E-3.

### Oblique cut

Considering that the quadrangle element is cut as represented in Figure 5.16 and the parameter $l$ represent the movement of the interface along a (1,1) direction, the velocity field corresponding to this variable can be chosen as:

$$\begin{aligned} V(x) &= \frac{r}{l}\sin(\theta), \quad V(y) = \frac{r}{l}\cos(\theta) \\ r &= \sqrt{x^2 + y^2} \end{aligned} \tag{5.68}$$

Using a finite difference scheme with a perturbation $\delta l$=$10^{-3}l$, we obtain the following values for the stiffness matrix derivatives:

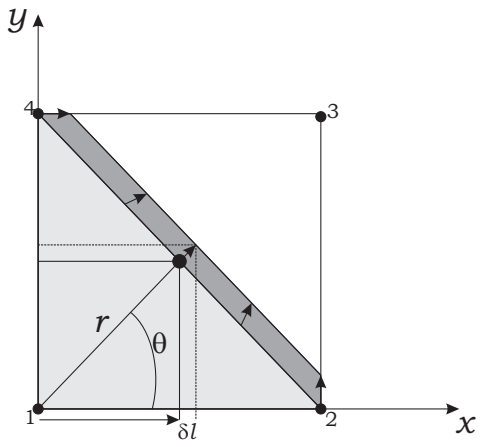



Figure 5.16: Perturbation of $l$ on the reference quadrangle

```
dk/dl=
      1,15367  0,32046 -0,76910  0,25640 -0,57692 -0,64098  0,19235  0,06411
      0,32046  1,15367  0,06411  0,19235 -0,64098 -0,57692  0,25640 -0,76910
     -0,76910  0,06411  1,15371 -0,64098  0,19231  0,25636 -0,57692  0,32050
      0,25640  0,19235 -0,64098  1,15381  0,06405 -0,76923  0,32052 -0,57692
     -0,57692 -0,64098  0,19231  0,06405  1,15385  0,32056 -0,76923  0,25636
     -0,64098 -0,57692  0,25636 -0,76923  0,32056  1,15385  0,06405  0,19231
      0,19235  0,25640 -0,57692  0,32052 -0,76923  0,06405  1,15381 -0,64098
      0,06411 -0,76910  0,32050 -0,57692  0,25636  0,19231 -0,64098  1,15371
```

and applying the analytical formula with the prescribed velocity field $V(x)$ as written in equation (5.68), we obtain the following numerical values:

```
dk/dl=
      1,15376  0,32047 -0,76917  0,25641 -0,57689 -0,64100  0,19230  0,06412
      0,32047  1,15376  0,06412  0,19230 -0,64100 -0,57689  0,25641 -0,76917
     -0,76917  0,06412  1,15377 -0,64100  0,19229  0,25640 -0,57689  0,32049
      0,25641  0,19230 -0,64100  1,15380  0,06410 -0,76921  0,32049 -0,57689
     -0,57689 -0,64100  0,19229  0,06410  1,15382  0,32050 -0,76921  0,25640
     -0,64100 -0,57689  0,25640 -0,76921  0,32050  1,15382  0,06410  0,19229
      0,19230  0,25641 -0,57689  0,32049 -0,76921  0,06410  1,15380 -0,64100
      0,06412 -0,76917  0,32049 -0,57689  0,25640  0,19229 -0,64100  1,15377
```

The Froebenius norm of this relative difference is 1.77E-3, which indicates that we obtain an excellent agreement between analytical and finite difference approaches.

## 5.10 Boundary integral method

In the previous section, the stiffness matrix derivative has been computed using a domain integral. However, given the analytical expression of the derivative:

$$\frac{dK}{dz} = \int_{\Omega(z)} div(B^T H B V)\, d\Omega$$

one can also apply the divergence theorem to obtain the following boundary integral formulation:

$$\frac{dK}{dz} = \int_{\Gamma(z)} B^T HBVn \; d\Gamma$$

Applied to a 1D problem, the stiffness matrix can be defined as:

$$\begin{aligned} k(x) &= B^T(x) HB(x) \\ K &= \int_0^l k(x) dx \end{aligned}$$

The derivative of this expression with respect to the parameter '$l$' gives:

$$\begin{aligned} \frac{dK}{dl} &= \lim_{\Delta l \to 0} \frac{1}{\Delta l} \left[ \int_0^{l+\Delta l} k(x)\, dx - \int_0^l f(x)\, dx \right] = \lim_{\Delta l \to 0} \frac{1}{\Delta l} \int_l^{l+\Delta l} k(x)\, dx \\ &= \lim_{\Delta l \to 0} \frac{1}{\Delta l} \left[ k(b)\, \Delta l + \mathcal{O}\left(\Delta l^2\right) \right] = k(l) \end{aligned}$$

and the sensitivity of the stiffness matrix is equal to the value of the function at $l$. The previous second degree 1D element example has the following analytical stiffness matrix:

$$\begin{aligned} k(x) &= \begin{bmatrix} \frac{16x^2}{L^4} - \frac{24x}{L^3} + \frac{9}{L^2} & \frac{16x^2}{L^4} - \frac{16x}{L^3} + \frac{3}{L^2} & -\frac{32x^2}{L^4} + \frac{40x}{L^3} - \frac{12}{L^2} \\ \frac{16x^2}{L^4} - \frac{16x}{L^3} + \frac{3}{L^2} & \frac{16x^2}{L^4} - \frac{8x}{L^3} + \frac{1}{L^2} & -\frac{32x^2}{L^4} + \frac{24x}{L^3} - \frac{4}{L^2} \\ -\frac{32x^2}{L^4} + \frac{40x}{L^3} - \frac{12}{L^2} & -\frac{32x^2}{L^4} + \frac{24x}{L^3} - \frac{4}{L^2} & \frac{64x^2}{L^4} - \frac{64x}{L^3} + \frac{16}{L^2} \end{bmatrix} \\ K &= \int_0^l k(x) dx \\ &= \begin{bmatrix} \frac{27L^2l - 36Ll^2 + 16l^3}{3L^4} & \frac{9L^2l - 24Ll^2 + 16l^3}{3L^4} & \frac{-36L^2l + 60Ll^2 - 32l^3}{3L^4} \\ \frac{9L^2l - 24Ll^2 + 16l^3}{3L^4} & \frac{3L^2l - 12Ll^2 + 16l^3}{3L^4} & \frac{-12L^2l + 36Ll^2 - 32l^3}{3L^4} \\ \frac{-36L^2l + 60Ll^2 - 32l^3}{3L^4} & \frac{-12L^2l + 36Ll^2 - 32l^3}{3L^4} & \frac{48L^2l - 96Ll^2 + 64l^3}{3L^4} \end{bmatrix} \end{aligned}$$

Analytically, the derivative is easily obtained:

$$\frac{dK}{dl} = \begin{bmatrix} \frac{9L^2 - 24Ll + 16l^2}{L^4} & \frac{3L^2 - 16Ll + 18l^2}{L^4} & \frac{-12L^2 + 40Ll - 32l^2}{L^4} \\ \frac{3L^2 - 16Ll + 18l^2}{L^4} & \frac{L^2 - 8Ll + 16l^2}{L^4} & \frac{-4L^2 + 24Ll - 32l^2}{L^4} \\ \frac{-12L^2 + 40Ll - 32l^2}{L^4} & \frac{-4L^2 + 24Ll - 32l^2}{L^4} & \frac{16L^2 - 64Ll + 64l^2}{L^4} \end{bmatrix} = k(l)$$

and we can remark that this expression is equal to the value of the $k(x)$ for $x = l$.

### Horizontal cut

The quadrangle problem introduced in the previous example (Fig. 5.15) can also be computed using a boundary integral:

$$\begin{aligned} \frac{dK}{dl} &= \int_{\Gamma(z)} B^T HBVn \; d\Gamma \\ &= \int_{\Gamma_1} B^T HBVn d\Gamma_1 + \int_{\Gamma_2} B^T HBVn d\Gamma_2 + \int_{\Gamma_3} B^T HBVn d\Gamma_3 + \int_{\Gamma_4} B^T HBVn d\Gamma_4 \end{aligned}$$

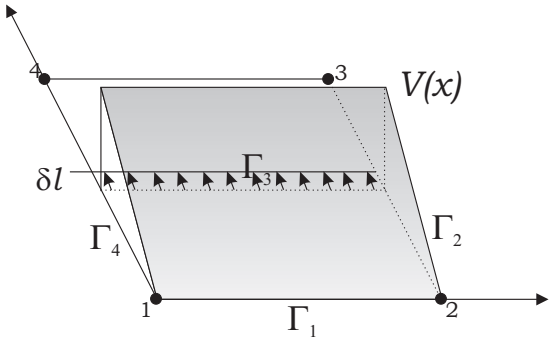



Figure 5.17: Boundary integration of the stiffness matrix derivative

Noticing in Figure 5.17 that the velocity is zero on $\Gamma_1$ and that the scalar product of the velocity field with the normal at the boundaries $\Gamma_2$ and $\Gamma_4$ is also null because the velocity field is aligned on the $y$ direction, this expression can be reduced to:

$$\frac{dK}{dl} = \int_{\Gamma_3} B^T H B V n d\Gamma_3$$

and the numerical integration of this expression result is in perfect agreement with the solution obtained with a domain integral in the previous section.

```
dk/dl=
      0.46467  0.24036 -0.27236  0.04807 -0.40064 -0.24037  0.20833 -0.048052
      0.24036  0.54485 -0.04807  0.12822 -0.24038 -0.32051  0.04810 -0.352564
     -0.27236 -0.04807  0.46467 -0.24036  0.20833  0.04805 -0.40064  0.240379
      0.04807  0.12822 -0.24036  0.54485 -0.04810 -0.35256  0.24038 -0.320512
     -0.40064 -0.24038  0.20833 -0.04810  0.46481  0.24040 -0.27250  0.048081
     -0.24037 -0.32051  0.04805 -0.35256  0.24040  0.54489 -0.04808  0.128185
      0.20833  0.04810 -0.40064  0.24038 -0.27250 -0.04808  0.46481 -0.240408
     -0.04805 -0.35256  0.24037 -0.32051  0.04808  0.12818 -0.24040  0.544891
```

**Oblique cut**

Considering the other test case with an oblique cut on the same quadrangular element and applying the boundary integral formulation with the velocity field of (5.68), the Froebenius norm of the relative difference 1.84E-3.

## 5.11 Bi-Material element

In the case of a bi-material element, the situation is different and it is more similar to a classical CAD shape problem because a part of the shape function involved in the approximation depends on the parameter '$z$'. Reminding the expression of the stiffness matrix:

$$K = \int_\Omega \begin{bmatrix} B_u \\ B_a \end{bmatrix} H \begin{bmatrix} B_u & B_a \end{bmatrix} d\Omega = \sum_{\Delta_i} \sum_{gpt}^{n_{gpt}} \begin{bmatrix} B_u \\ B_a \end{bmatrix} H \begin{bmatrix} B_u & B_a \end{bmatrix} |J_1| \, |J_2| \, w_{gpt}$$

Derivating this expression with respect to a design parameter $z$, we have:

$$\begin{aligned}\frac{\partial K}{\partial z} &= \sum_{\Delta_i}\sum_{gpt}^{n_{gpt}} w_{gpt}\begin{bmatrix}\frac{\partial B_u}{\partial z}\\ \frac{\partial B_a}{\partial z}\end{bmatrix} H\left[B_u\ B_a\right]|J_1|\,|J_2| \\ &+\sum_{\Delta_i}\sum_{gpt}^{n_{gpt}} w_{gpt}\begin{bmatrix}B_u\\ B_a\end{bmatrix} H\begin{bmatrix}\frac{\partial B_u}{\partial z} & \frac{\partial B_a}{\partial z}\end{bmatrix}|J_1|\,|J_2| \\ &+\sum_{\Delta_i}\sum_{gpt}^{n_{gpt}} w_{gpt}\begin{bmatrix}B_u\\ B_a\end{bmatrix} H\left[B_u\ B_a\right]|J_1|\,\frac{\partial\,|J_2|}{\partial z}\end{aligned}$$

The term $\partial B_u/\partial z$ is equal to zero because the position of the material interface has no effect on the standard part of the element approximation. Reminding the form of the approximation:

$$u = \sum N_i u_i + \sum N_i F_i a_i$$

The derivative of $B_a$ with $z$ for node $r$ gives:

$$\begin{aligned}\frac{\partial B_{ar}}{\partial z} &= \begin{bmatrix}\frac{\partial}{\partial z}(N_{r,x}F_r + N_r F_{r,x})\\ \frac{\partial}{\partial z}(N_{r,y}F_r + N_r F_{r,y})\end{bmatrix} = J_1^{-1}\begin{bmatrix}\frac{\partial}{\partial z}(N_{r,\xi}F_r + N_r F_{r,\xi})\\ \frac{\partial}{\partial z}(N_{r,\eta}F_r + N_r F_{r,\eta})\end{bmatrix} \\ &= J_1^{-1}\begin{bmatrix}\frac{\partial N_{r,\xi}}{\partial z}F_r + N_{r,\xi}\frac{\partial F_r}{\partial z} + \frac{\partial N_r}{\partial z}F_{r,\xi} + N_r\frac{\partial F_{r,\xi}}{\partial z}\\ \frac{\partial N_{r,\eta}}{\partial z}F_r + N_{r,\eta}\frac{\partial F_r}{\partial z} + \frac{\partial N_r}{\partial z}F_{r,\eta} + N_r\frac{\partial F_{r,\eta}}{\partial z}\end{bmatrix}\end{aligned}$$

To clarify this expression, let's limit the expression to 1D example. Using the chain rule of derivatives, one gets:

$$\begin{aligned}\frac{\partial B_{ar}}{\partial z} &= J_1^{-1}\begin{bmatrix}\frac{\partial N_{r,\xi}}{\partial z}F_r + N_{r,\xi}\frac{\partial F_r}{\partial z} + \frac{\partial N_r}{\partial z}F_{r,\xi} + N_r\frac{\partial F_{r,\xi}}{\partial z}\end{bmatrix} \\ &= J_1^{-1}\left[\frac{\partial N_{r,\xi}}{\partial \xi}\frac{\partial \xi}{\partial z}F_r + \frac{\partial N_r}{\partial \xi}\frac{\partial \xi}{\partial z}F_{r,\xi} + N_{r,\xi}\frac{\partial F_r}{\partial z} + N_r\frac{\partial F_{r,\xi}}{\partial z}\right]\end{aligned}$$

where the two first terms are similar to the void approach except the presence of the enrichment function $F_r$ and its derivative $F_{r,\xi}$. The last two terms takes into account of the modification of the enrichment function $F_r$. In the case of 1D bi-material element, $F_r$ is explicit in terms of $l$ as shown in chapter 3.

Chapter 6

# Numerical applications

Various mechanical responses can be chosen as objective functions and/or design restrictions: compliance, all stress components, Von Mises stress, displacements, eigenfrequency, mass and geometric characteristics such as the volume. Presently, the implementation of the X-FEM is available for 1, 2 and 3D elements with approximation orders up to the third degree for both void-material and bi-material interfaces. The Level Set description can be defined in different ways within a dedicated "Level Set Modeler" developed into the C++ software OOFELIE. Level Sets entities can be constructed from standard shape primitives (circle, quadrangles, ellipses, ...), quadrics or from a given set of points and curves. Complex curves and surfaces can be represented using a NURBS converted into an associated Level Set object. Finally, complex native CSG Level Set models can be constructed within OOFELIE or imported from the Gmsh [75] native Modeler or from generic STEP files. However, for these geometries, no optimization is possible. Currently, Gmsh is not able to recognize the geometrical parameters of the entities in CAD files.

An academic version of the CONLIN optimizer described in [65] has been linked to OOFELIE and the optimization module initiated by Adam [4] has been widely extended to treat X-FEM shape optimization applications. Access to a larger choice of optimization algorithms has been made available by coupling OOFELIE to BOSS Quattro [142] using a dedicated driver. In all the following test cases, the CONLIN optimizer is used unless explicitly mentioned.

## 6.1 Validation example: the plate with a hole

The first application presented is the academic benchmark of the plate with a hole under a biaxial traction far field $\sigma_x, \sigma_y$ (see Fig. 6.1). Stress concentrations occur along the hole boundaries and the objective is to obtain a new geometry that minimizes these stress concentrations. Ideally the geometry should present a uniform stress field along the hole. In [147], the analytical result shows that, in an infinite plate under bi-axial traction field, the optimal hole is elliptical, the geometry being characterized by a ratio $a/b$ between the two major/minor axes of the ellipse that is equal to $k{=}\sigma_x/\sigma_y$. In this case, the tangential stress is uniform along the ellipse lips.

For a unidimensional far field $\sigma = p$ along a direction $(\cos\alpha, \sin\alpha)$ with respect to the axis $x$,

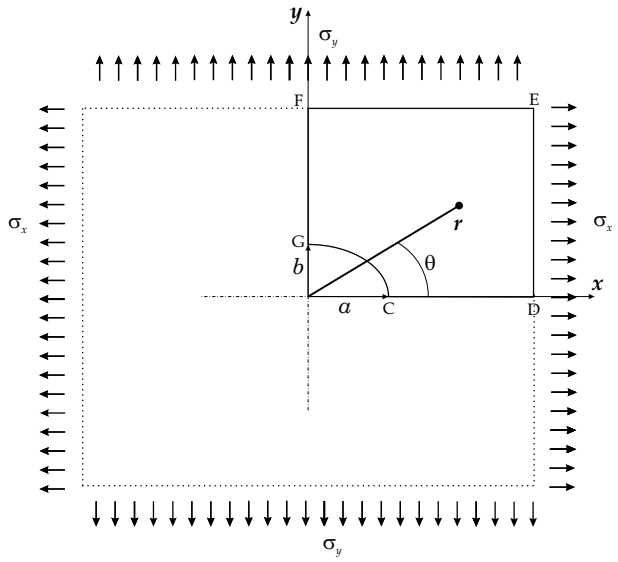



Figure 6.1: Geometry of the plate with a hole

the tangential stress along the lips of the elliptical hole is given by [147]:

$$\sigma_\theta = p\frac{(1+k)^2\sin^2(\alpha+\theta) - \sin^2\alpha - k^2\cos^2\alpha}{\sin^2\theta + k^2\cos^2\theta}$$

where $\theta$ is the angular position from the $x$ axis. In the case of bi-axial traction field $(\sigma_1, \sigma_2)$, the total stress field is given by the superposition of the two contributions $\sigma_1 = \sigma_x(\alpha = 0)$ and $\sigma_2 = \sigma_y(\alpha = \frac{\pi}{2})$:

$$\sigma_\theta = \sigma_1\frac{(1+k)^2\sin^2\theta - k^2}{\sin^2\theta + k^2\cos^2\theta} + \sigma_2\frac{(1+k)^2\cos^2\theta - 1}{\sin^2\theta + k^2\cos^2\theta}$$

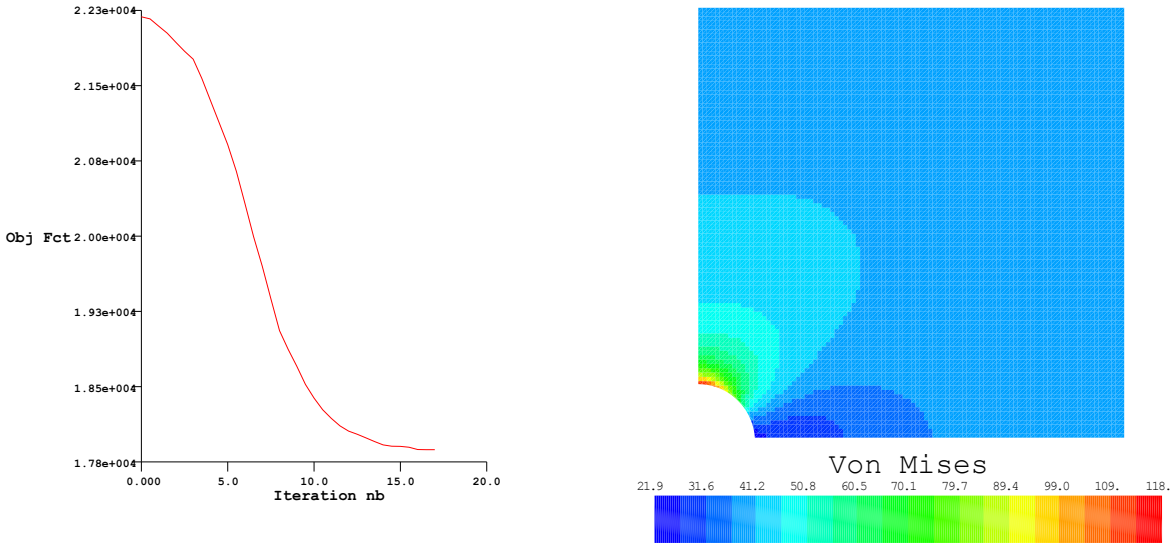
If $k$=$\sigma_x/\sigma_y$, one obtains a uniform circumferential stress along the hole which is equal to:

$$\sigma_\theta = \sigma_x\frac{(1+k)^2\sin^2\theta + k(1+k)^2\cos^2\theta - k^2 - k}{\sin^2\theta + k^2\cos^2\theta} = (1+k)\sigma_x \tag{6.1}$$

From equation (6.1), the general optimization problem can be formulated as finding the shape which presents a uniform circumferential stress along the hole. In practice, the problem is equivalently stated as minimizing the volume of the plate with a constraint on the maximum circumferential stress.

Due to symmetry conditions, only one quarter of the geometry can be considered by applying suitable boundary conditions as depicted in Figure 6.1. A uniform traction $\sigma_x$=45 MPa is applied on the right side and another one of value $\sigma_y$=22.5 MPa along the $y$ axis for the upper side. The plate is placed in plane stress state. The Young modulus is $E$=210 Mpa and Poisson's ratio $\nu$=0.3. The elements are first degree quadrangle with regular grid of 50×50 elements. The analysis domain is a square of dimension 1.5 m × 1.5 m. Following Liefooghe and Fleury [101], we do not introduce a constraint on the tangential stress but rather formulate the problem under an equivalent form: minimize the volume of the plate with a constraint on the maximum Von Mises stress. As the radial stress is zero along the ellipse free boundary, the Von Mises stress is equal to the tangential stress. The upper bound on the maximum Von Mises stress $\sigma_{vm}$ is set to

95 MPa. The design variables are the two axes of the ellipse $a$ and $b$. Their initial values are 0.2 and the side constraints are $\underline{x}_i$=1E-3 and $\overline{x}_i$=1.2. For this load case, the optimal ratio $k$ should be equal to 2.




(a) History of the volume objective function

(b) Initial shape with Von Mises distribution

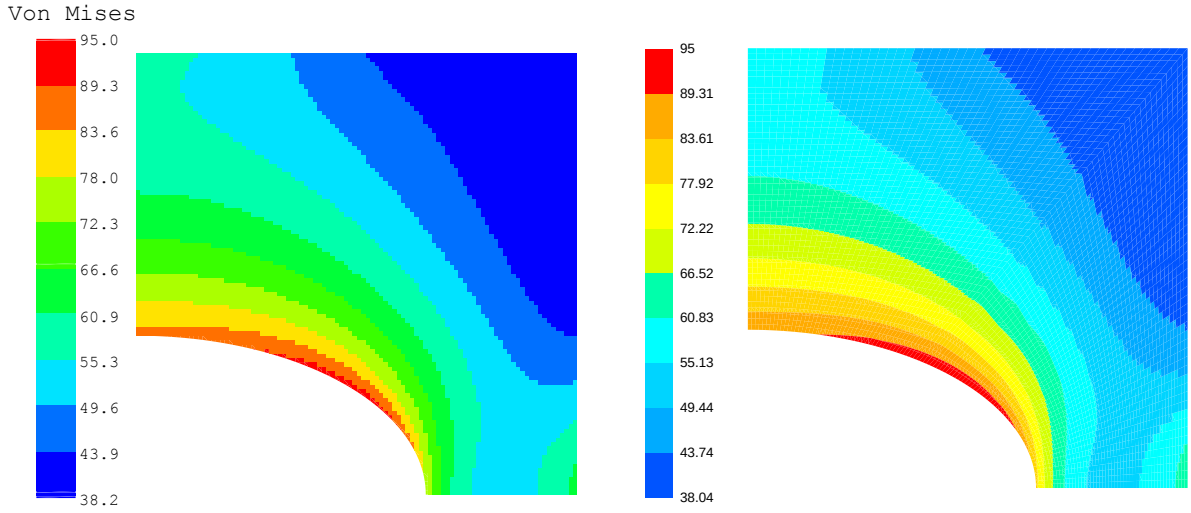
Figure 6.2: Optimization of a plate with a hole

The convergence criterion is based on the relative variation of the objective function: $\Delta f \leq$ 1E-3. It is satisfied after 17 iterations as depicted in Figure 6.2 (a). In fact, the number of iterations is function of the move limits applied to design variables (here 10% of the initial value). For instance, with 50% move limits, the solution is obtained after 7 iterations only. In Figure 6.2 (b) we can observe that in the initial configuration, the constraint on the Von Mises stress is violated. The optimizer first reduces the value of variables $a$ and $b$ in order to satify this constraint while increasing the total volume. When an admissible point is obtained, the optimizer modifies the parameters to minize the volume while satisfying the constraint on the Von Mises stress until an optimal point is attained with a sequence of steadily feasible designs.

As expected, we get a solution with a ratio of the ellipse axis close to the optimal value of 2 with parameter values $a$=9.7, $b$=5.3 and a ratio $k$=1.83. The optimal ratio is not achieved because of the effect of the finite dimension of our structure. We can remark in Figure 6.3 (a) that the hole is very close to the right edge of the plate.

For the sake of comparison, the same problem has been solved using Samcef and BOSS Quattro with a Finite Difference approach in order to compare X-FEM solution with a validated commercial tool. As we can clearly see in Figure 6.3, both solutions are very close as the $k$ ratios are equal to 1.83 and 1.85 respectively for X-FEM and FEM.

The same problem was studied by Tekkaya and Güneri [5] using Biological Growth Method and FEM. In Ref. [5], a reference stress value corresponding to the nominal Von Mises stress in the plate far from the hole is set to $\sigma_{ref}$=40 Mpa. The evolution of the Von Mises stress

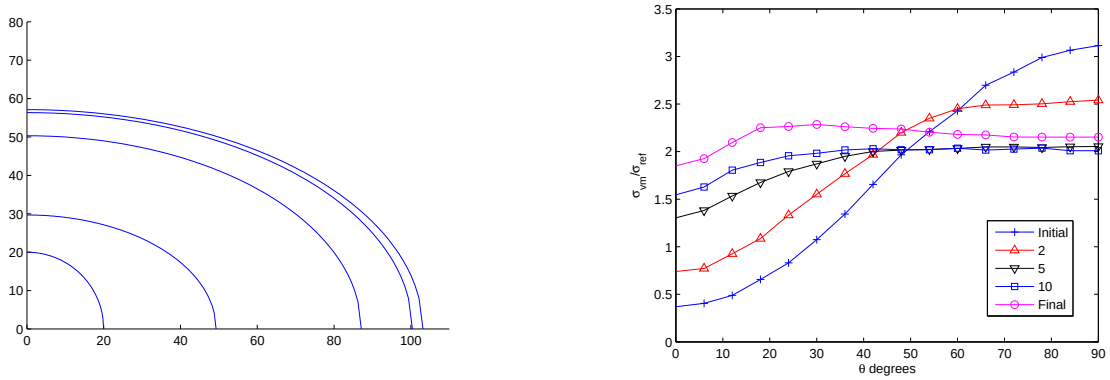



(a) Optimized configuration obtained with X-FEM; $k$=1.83 with Von Mises stresses

(b) Solution obtained with SAMCEF/BOSS Quattro; $k$=1.85 with Von Mises distribution

Figure 6.3: Final geometry with X-FEM and FEM

$\sigma_{vm}$ normalized with this reference stress $\sigma_{ref}$ is represented in Figure 6.4 (b) as a function of the angular position $\theta$ along the hole boundary. We can see that for the initial geometry, a stress concentration occurs at $\theta = 90°$ which corresponds to the point G in Figure 6.1 while the minimum is located at $\theta = 0°$ (Point C).

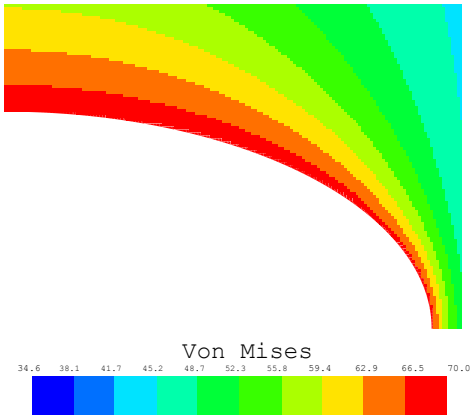



(a) Evolution of the ellipse boundary

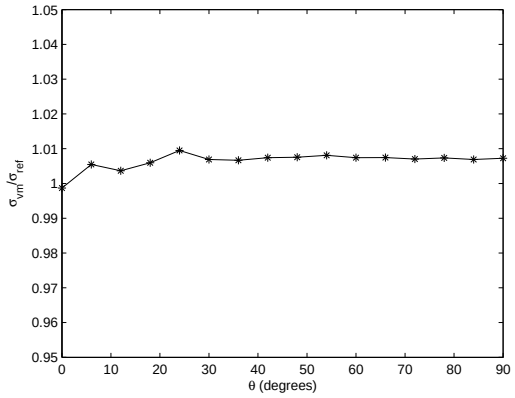
(b) Evolution of Von Mises stress along the ellipse boundary for different iterations

Figure 6.4: Evolution of the $\sigma_{vm}$ along the hole

According to the analytical solution, the stress concentration is around 3. We can see in Figure 6.4 (b) that during the optimization process, the optimizer successively modifies the shape of the hole to come to a new geometry, which presents a lower stress peak and a lower variation between highest and lowest stress. At the optimum, the normalized Von Mises stresses should be uniform along the ellipse boundary as predicted by the theory. The stress field along the hole is not perfectly uniform as one can notice in Figure 6.4. However, we can clearly see that stress concentration has been strongly reduced. The evolution from a circular to the optimal elliptical shape is shown in Figure 6.4.




(a) Solution obtained with an extended domain and a fine mesh around boundary. $k$=1.96. Von Mises plot




(b) $\sigma_{VM}/\sigma_{ref}$ around the ellipse boundary at the optimum

Figure 6.5: Solution with a larger analysis domain (4.5 m × 4.5 m)

This problem has been replayed on a rather larger domain (4.5 m × 4.5 m) to get closer to an infinite loading condition. We have set a lower max Von Mises stress which has the effect of keeping the optimal size of the hole rather small with respect to the dimensions of the plate. As the optimization is based on the stresses, we also create a very fine mesh close to the hole to obtain accurate Von Mises stress values and a fine geometrical representation of the hole as it should be very small with respect to the plate size. Doing so, we obtain the results presented in Figure 6.5 (a) and (b) where the stress is close to a constant value along the ellipse boundary. The variation between maximal and minimal stress values is now smaller than 1%.

The same problem could also be stated as the minimization of the volume with a restriction on the compliance. This formulation is quite unusual because the compliance, which is the inverse of the rigidity, is difficult to interpret unless it is expressed relatively to a reference value for the structure. For instance, the first problem treated in this section gives a $k$ ratio equal to 1.95 when the compliance is constrained to remain below its initial value. In this case, only one restriction is present, the problem is much simpler for the optimizer and only 5 iterations are necessary.

## 6.2 The 2D fillet in tension

The second numerical application is inspired by the investigation of stress concentration factors by Peterson [138] and revisited by Pedersen [137] as a shape optimization problem. The design problem consists in studying the effect of the connection zones in the 2D structure of Figure 6.6 and reducing the stress concentration factor. In the initial geometry, the parameters are fixed to the values:

- $l_1$=30 mm, $l_4$=60 mm, $l_2$=30 mm, $l_3$=90 mm;
- $l_f$=30 mm for applications in sections 6.2.1 and 6.2.2 while it is a design variable in the section 6.2.3.

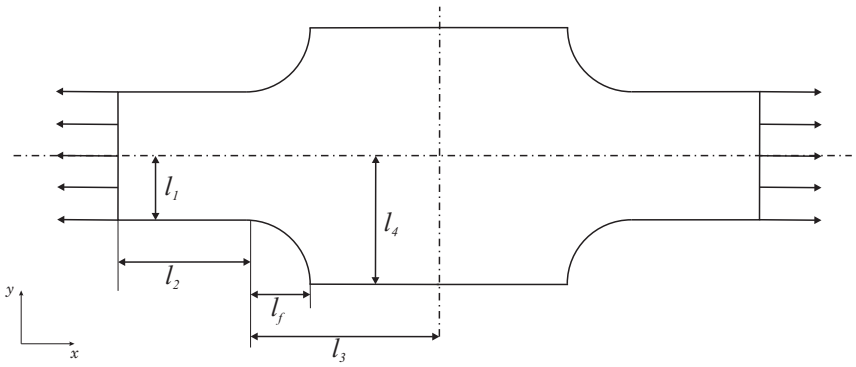
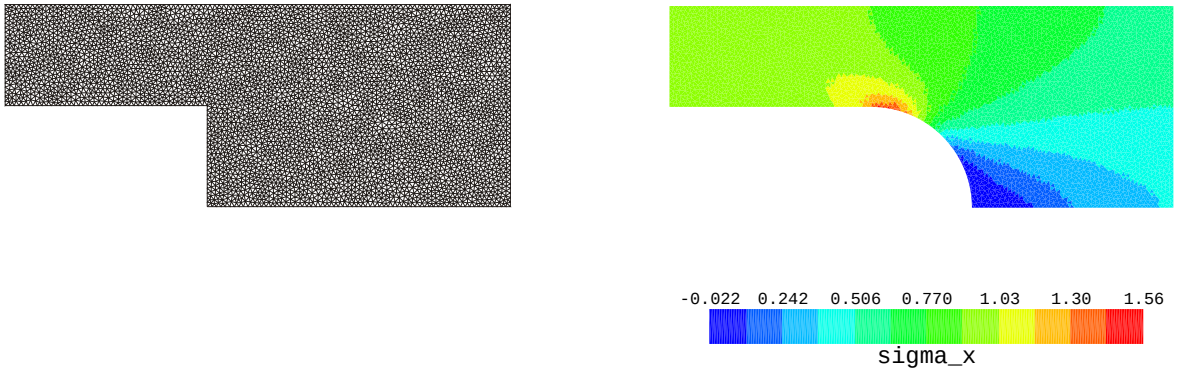



Figure 6.6: 2D fillet geometry and load case

A plane stress state is assumed and a uni-axial stress field $\sigma_x$=1 N/mm$^2$ is applied on the two ends. The normalized material properties are: Young's modulus $E = 1$ N/mm$^2$, Poisson's ration $\nu$=0.3. Due to symmetry conditions, only a quarter of the part is studied. The initial configuration is depicted in Figure 6.7 (b) while the mesh used for the following test cases is represented on the left side Figure 6.7 (a).




(a) Mesh used for all the test cases

(b) Initial shape: $\eta$=2, $\sigma_{max}$=1.563 N/mm$^2$, with Von Mises distribution

Figure 6.7: Mesh and X-FEM analysis of the initial geometry

### 6.2.1 Super circular connection

As pointed out by Pedersen in [137], a circular connection to a straight domain is not the best solution and it is better to use a super-circular connection which is defined by:

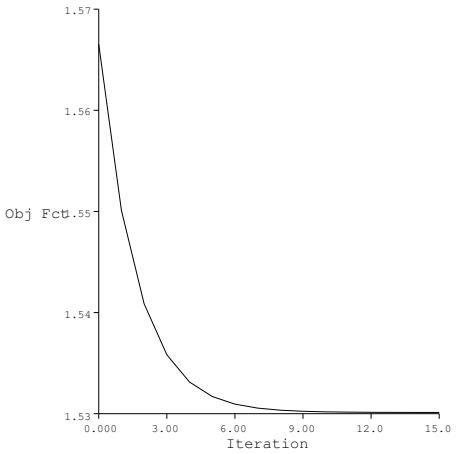
$$x_1^\eta + x_2^\eta = r$$

The goal of the optimization problem is to find the configuration presenting the minimum stress concentration:

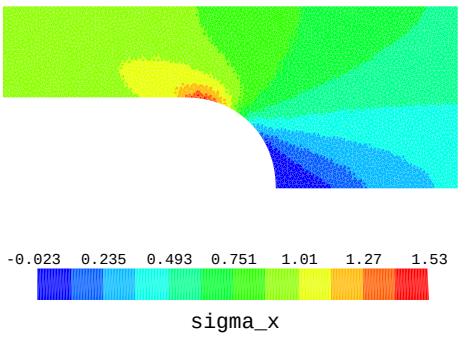
$$\min(\max_e \sigma_{x_e}^2)$$

where $e$ is the index of the finite elements. In practice, a multi objective approach is used by taking into account only the most critical elements, i.e. the top 10 % of the most stressed finite elements. No other constraint is prescribed except the side-constraints over the variable $\eta$ within

the range $2 \leq \eta \leq 4$. The initial configuration, with $\eta$=2, presents a stress concentration of $\sigma^{\max}/\sigma^{\infty} = 1.56253$ with the mesh given in Fig. 6.7.




(a) History of the maximum stress




(b) Final shape: $\eta$=2.09, $\sigma_{max}$=1.526 N/mm$^2$ with Von Mises distribution

Figure 6.8: Optimization with a super circle

The optimum, obtained after ten iterations, provides a rather limited improvement as the stress concentration is reduced to the value of $\sigma^{\max}/\sigma^{\infty} = 1.52612$ (small gain of 2.75%) and an optimum value of $\eta$=2.0935 (Fig. 6.8).

### 6.2.2 Generalized super circular connection

The previous test case shows that only a minor improvement can be obtained by using a super circle in comparison to a circle. Therefore, we replace the super circle fillet with a generalized super circle defined as:

$$x_1^{\eta} + x_2^{\xi} = r$$

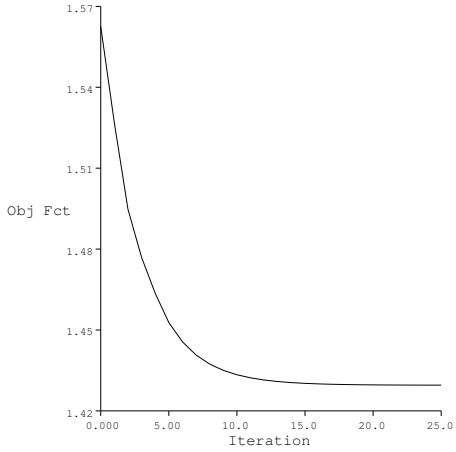
The problem is the same as the previous one, but it handles now two variables ($\xi$ and $\eta$) which are both constrained between the values 2 and 4. As it can be noticed, the additional exponent allows a deepest modification of the fillet curvature, which leads to a more significant reduction of 6.7 % of the stress concentration (Fig. 6.9).

### 6.2.3 Generalized super ellipse connection

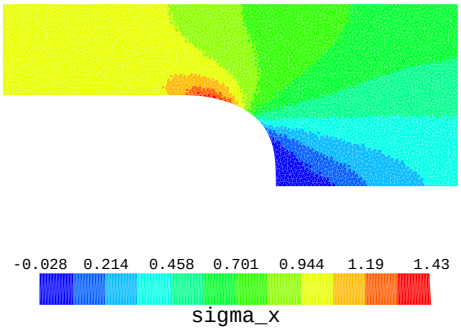
The last test case shows the largest impact of the fillet length on the stress concentration. A generalized super ellipse is used to define the boundary of the fillet:

$$\left|\frac{x_1}{a}\right|^{\eta} + \left|\frac{x_2}{b}\right|^{\xi} = r$$

With an unlimited fillet length the stress concentration could be canceled. Hence, the optimization variable $a$ is constrained between the values 20 and 60 (initial value of 30). The other variables are constrained between $2 \leq \xi \leq 4$ and $2 \leq \eta \leq 4$. As expected, the optimization yields



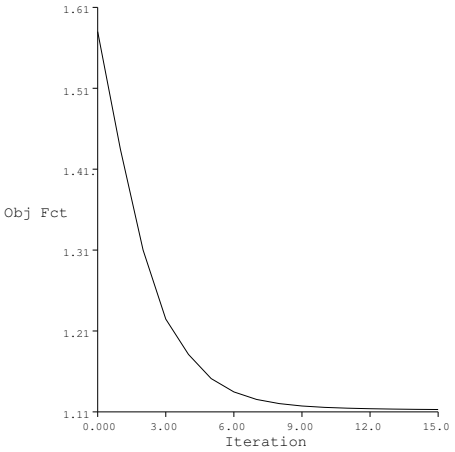

(a) History of the maximum stress



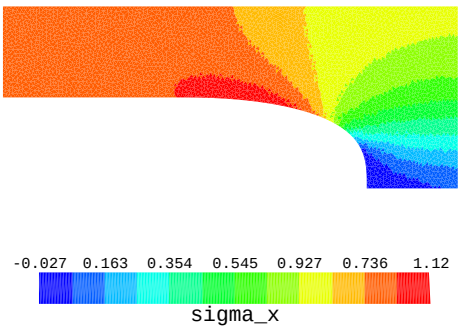

(b) Final shape: $\xi$=3.26, $\eta$=2.09, $\sigma_{max}$=1.43 N/mm$^2$ with Von Mises distribution

Figure 6.9: Optimization with a generalized super circle

the same $\xi$ and $\eta$ obtained with the previous optimizations and it maximizes the value of the parameter $a$ representing the fillet length. The stress concentration has been reduced by 22% with respect to the previous case and by nearly 30% to the initial circular fillet (Fig. 6.10).




(a) History of the maximum stress




(b) Final shape: $\xi$=3.26, $\eta$=2.09, $a$=60, $\sigma_{max}$=1.12 N/mm$^2$ with Von Mises distribution

Figure 6.10: Optimization with a generalized super ellipse

## 6.3 Rotating disc of uniform strength

In applications characterized by high rotation speeds such as turbine and compressor discs, it is often desirable to design the rotors for constant stress conditions. In this type of components, the main loads are the centrifugal forces and the corresponding stresses are therefore symmetric with respect to the rotation axis.

The rotating disc of uniform strength is a disc for which radial and hoop stresses are equal and constant for all values of the radius $r$:

$$\sigma_r = \sigma_t = \sigma,\ 0 \leq r \leq R$$

$R$ being the external radius of the disc. This problem has been treated by Braibant [37] using a FEM shape optimization. The expression of the analytical thickness of the disc can be found in [170]:

$$h = h_R\, e^{\frac{1-x^2}{2s}} \tag{6.2}$$

where $x = r/R$, $s = \sigma/(\rho\omega^2 R^2)$ and $h_R$ the thickness at radius $r = R$. In practice, the thickness $h_R$ is imposed by the loads generated by the blades acting on the outer radius of the disc to obtain a given stress condition $\sigma = \frac{F}{2\pi R h_R}$. Given the thickness $h_R$ and $R$, one yields a theoretical thickness law represented in the Figure 6.11.

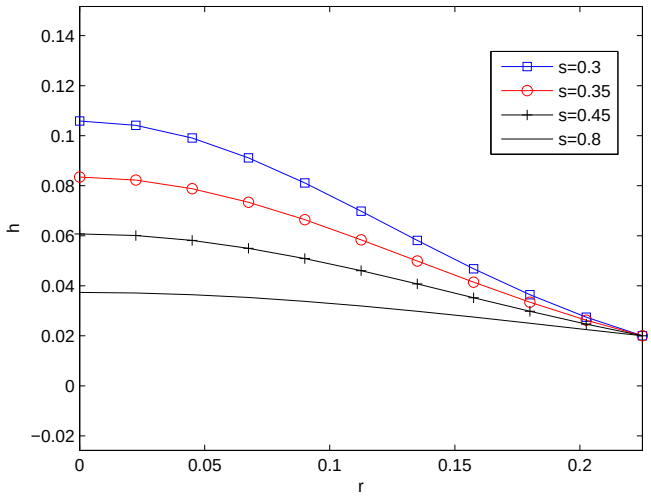



Figure 6.11: Theoretical thickness of the disc for different values of $s$. $R$=0.225 m, $h_R$=0.02 m

The axisymmetric model is given in Fig. 6.12. Because of symmetry conditions, half of the profile is studied. The boundary conditions are the following: the displacements are blocked along the radial direction for the nodes located on the axis of rotation (line a-d in Figure 6.12) and the vertical displacements are blocked on the symmetry plane (line a-b in Figure 6.12). Two loads are considered simultaneously: the centrifugal loads coming from a rotation speed of 2000 rad/s and a radial load, applied at the radius $r = R$, of 60 daN. The outer radius $R$ is fixed at 225 mm and the initial height is equal to 30 mm. The upper face of the disc is represented with a 5 control

points NURBS curve. The design variables are the positions along the $z$ axis of the 5 control points (represented with black dot in Figure 6.12). The aim of this optimization problem is to design a disc of minimal mass with a stress constraint of $\sigma$=60 daN/mm$^2$ imposed for both $\sigma_r$ and $\sigma_t$. The density of the material is $\rho$=8000 kg/m$^3$, the Young modulus is $E$=194E3 N/mm$^2$ and the Poisson's ratio $\nu$ is 0.3.

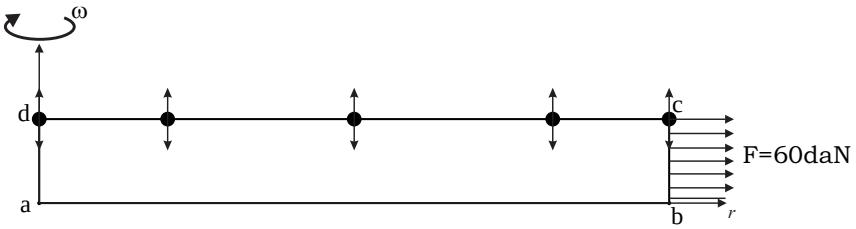
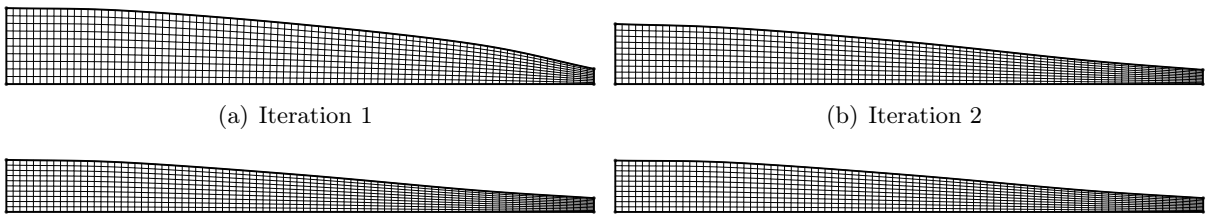



Figure 6.12: Initial geometry of the disc, load case and design variables

To compare FEM and X-FEM solutions, we carry out a FEM shape optimization on a grid composed of 87×10 first order axisymmetric quadrangle element. The optimization is realized using BOSS Quattro with a FD sensitivity analysis. The Figure 6.13 illustrates the evolution of the disc shape and the related transfinite mesh.

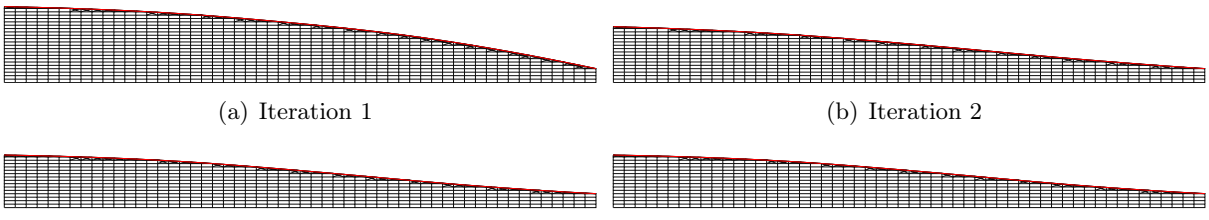


(a) Iteration 1 (b) Iteration 2

(c) Iteration 3 (d) Optimal geometry

Figure 6.13: History of the disc thickness with FEM and BOSS Quattro

The same study is realized with bi-linear quadrangle axisymmetric X-FEM elements and a mesh grid shown in Figure 6.14.



(a) Iteration 1 (b) Iteration 2

(c) Iteration 3 (d) Optimal geometry

Figure 6.14: History of the disc shape with X-FEM

Comparing Figures 6.13 and 6.14 representing the history of the thickness of the disc, we can see that both methods converge to similar geometries. Indeed, the Figure 6.15 shows that the two approaches reach a nearly same optimal solution requiring both 5 iterations to convergence.

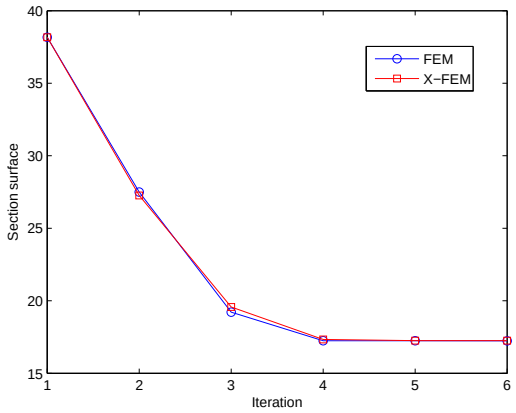


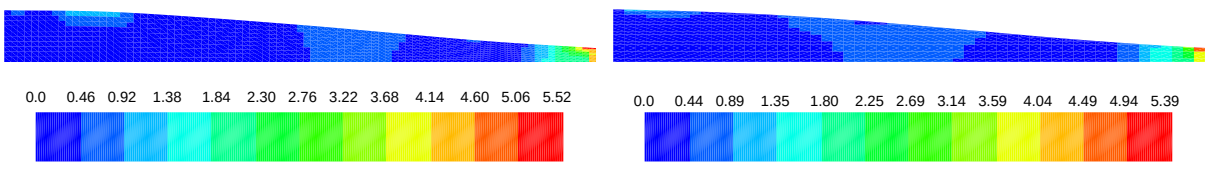

Figure 6.15: Comparison of the objective function history for the FEM and X-FEM

| $x$ | $h$ *FEM* | $h$ *X-FEM* | $h$ *Theory* | *X-FEM Rel. error* |
|---|---|---|---|---|
| 1 | 5.29 | 5.29 | 5 | 5.8 |
| 0.9 | 6.56 | 6.57 | 6.46 | 1.7 |
| 0.8 | 8.21 | 8.22 | 8.13 | 1.1 |
| 0.7 | 10.09 | 10.09 | 9.95 | 1.4 |
| 0.6 | 12.09 | 12.13 | 11.86 | 2.27 |
| 0.5 | 14.09 | 14.11 | 13.76 | 2.54 |
| 0.4 | 15.97 | 15.94 | 15.54 | 2.57 |
| 0.3 | 17.60 | 17.52 | 17.08 | 2.57 |
| 0.2 | 18.85 | 18.66 | 18.27 | 2.13 |
| 0.1 | 19.58 | 19.44 | 19.03 | 2.15 |
| 0 | 19.67 | 19.6 | 19.29 | 1.60 |

Table 6.1: Comparison of the thickness of the disc with theoretical solution (mm)

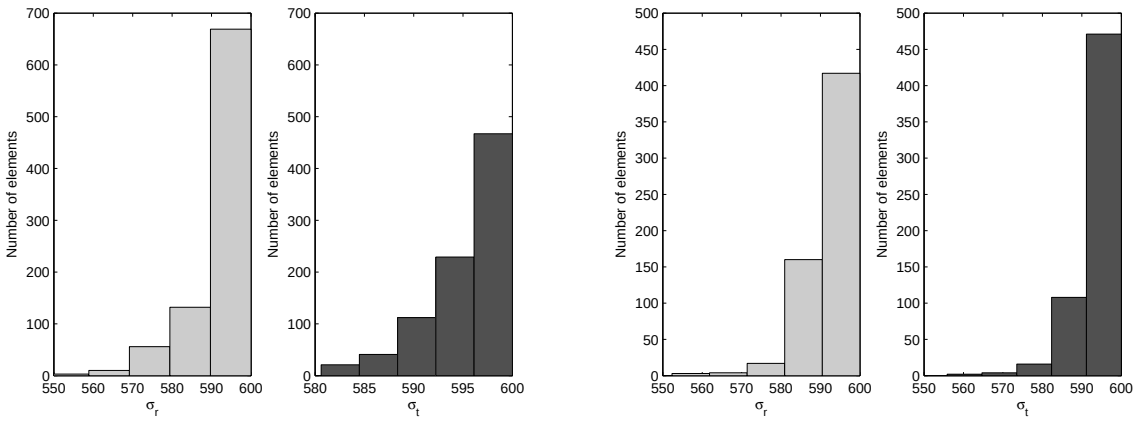
The Table 6.1 compares the solutions obtained with FEM, X-FEM and the theoretical values for the thickness of the disc. FEM and X-FEM agree very well with the theoretical results as the relative difference is of the order of 3% except at the outer radius of the disc where the difference is about 6 %.

The results of the Table 6.1 are confirmed by the Figure 6.16 representing the relative difference between the radial and the hoop stresses for FEM and X-FEM. According to the error values in the Table 6.1, we can observe that hoop and radial stresses are very close. The relative difference is lower than 1 % in a large part of disc. However, at vincinity of the outer radius, the uniform strength state is not met. The deviation between hoop and radial stresses being in the order of 5 %. This can be explained by the fact that the uniform strength theoretical solution assumes that $\sigma_z$ is equal to zero in the disc. In practice, this is not the case as the disc geometry is not very thin. Moreover, FEM and X-FEM solutions do not present a strictly constant radial and hoop stress as illustrated by the histograms in Figure 6.17. Both methods converged to a geometry where $\sigma_r$ and $\sigma_t$ present a maximum value of 600 N/mm$^2$, X-FEM solution presenting a mean value $\bar{\sigma_r}$=592 N/mm$^2$ and $\bar{\sigma_t}$=594 N/mm$^2$. These numbers are very close for the FEM model: $\bar{\sigma_r}$ = 592 N/mm$^2$ and $\bar{\sigma_t}$=595 N/mm$^2$.




(a) FEM (b) X-FEM

Figure 6.16: Relative difference of radial and hoop stresses (%)




(a) FEM (b) X-FEM

Figure 6.17: Histogram of radial and hoop stresses for FEM and X-FEM model

## 6.4 Shape and topology optimization of a 2D bracket support

This classic shape optimization problem has been initially proposed by Bennet and Botkin in [30] and revisited by many other researchers (e.g. [78, 183]). The bracket support, represented in Figure 6.18, is submitted to a vertical bending load $F$=15 kN applied at the left hole $C_1$, while the two other ones are clamped. The load is applied at point A, the center of circle $C_1$ being completely filled with a stiff material. The objective is to minimize the mass of the structure with a constraint on the Von Mises stress.

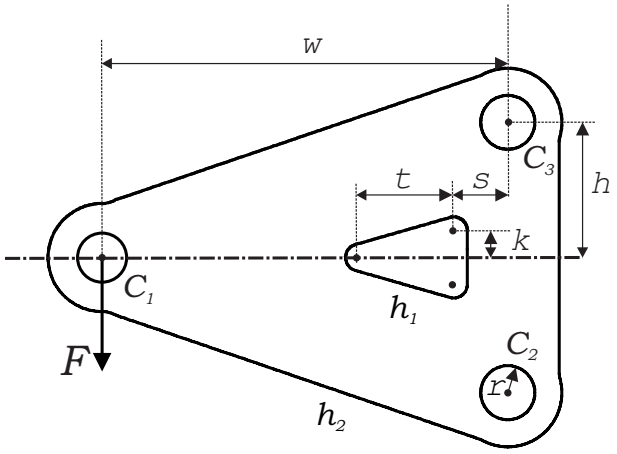
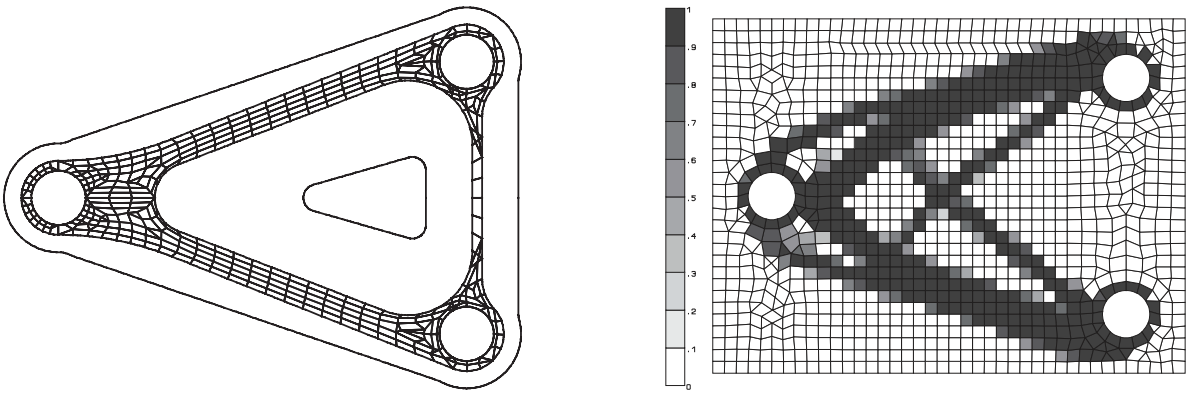



Figure 6.18: 2D bracket support geometry

This problem has been widely presented by several researchers to validate and compare numerical analysis methods. The two Figures 6.19 illustrate the optimal shape obtained by Zhang [183] using a shape optimization method and the same problem studied by Duysinx [57] using a topology optimization approach.

Comparing the two solutions, we can observe some major differences. The shape optimization is not able to modify the topology of the structure and the optimal solution is similar to the



(a) Shape optimization solution obtained by Zhang [183]

(b) Topology optimization solution obtained by Duysinx [57]

Figure 6.19: Two optimal geometries for the bracket

initial geometry while being much lighter. Conversely, the topology optimization is free to create holes to propose a new connectivity and a novel shape. Hence, we can observe that the very thin structural part between circle $C_2$ and $C_3$ present in the shape optimization solution has been completely removed in the topology optimized geometry. Therefore, the topology can use this material saving to strengthen the two arms and enlarge the overall thickness by placing secondary arms. Moreover, we can remark that both methods tend to a geometry where the two arms joining $C_1$ to $C_2$ and $C_1$ to $C_3$ have a higher angle, moving the supports to a new position respectively higher and lower for the circles $C_3$ and $C_2$.

From these observations, it is clear that if the shape optimization would be able to remove the part joining circle $C_2$ and $C_3$, a better intermediate geometry between classical shape and topology solutions could be obtained. To illustrate the capability of the present method to achieve such modification of the topology, we consider different formulations and geometrical parameterizations in the following.

For all the test cases, the material is characterized by a Young modulus of $E$=207.4E3 N/mm$^2$, a Poisson coefficient $\nu$=0.3. Plane stress state is assumed. The geometrical parameters present in Figure 6.18 are $w$=150 mm, $r$= 10mm and $h$= 50 mm whereas the other parameters $t$, $s$, $k$ are given in the relevant section.

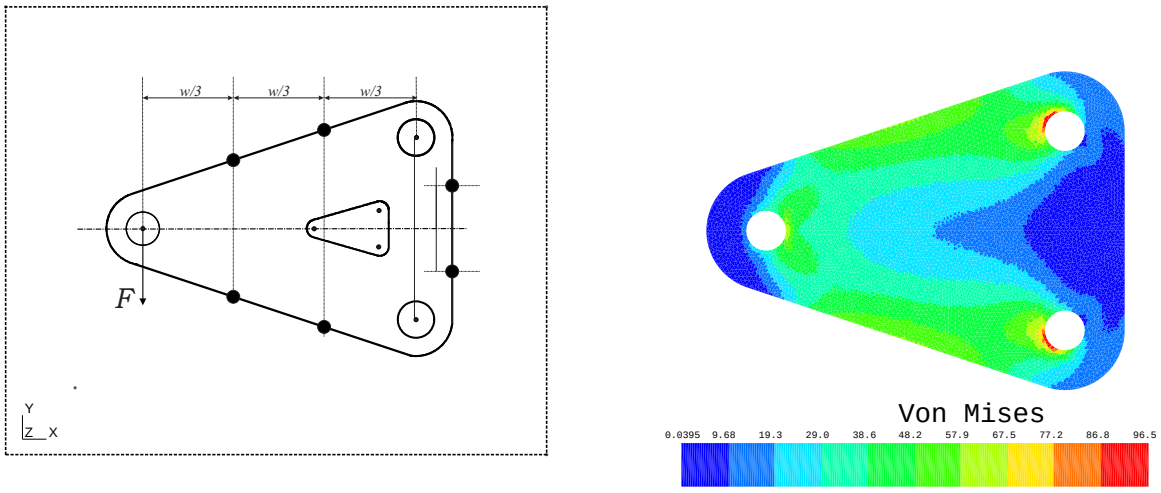
### 6.4.1 Minimization of the volume with stress constraints

First, we investigate the formulation generally adopted for the shape optimization method that consists in minimizing the volume with restrictions on the maximum Von Mises stress: $\sigma_{eq} \leq 120$ N/mm$^2$ .

#### Simplified geometry

The initial geometry of the bracket is simplified by removing the inner hole $h_1$ (see Fig. 6.18). The optimization is realized by modifying solely the external boundary $h_2$. It is modeled with 3 circle arcs and 3 line segments. Each segment is defined with a 6 control points NURBS curve.

The opening angles of the 3 circle arcs are fixed and a tangency condition between the circles and the connected segments is imposed. From the 18 control points (represented by black dots in Figure 6.20 (a)), 6 control points remain independent and are considered as design variables as well as the 3 radii of the circle arcs. For the vertical piece of line, the control points are allowed to move only along the $x$ direction whereas the control points of the two remaining piece of lines evolves along the $y$ axis.




(a) Analysis domain and parametrization of external boundaries

(b) Initial shape with Von Mises distribution ($N/mm^2$)

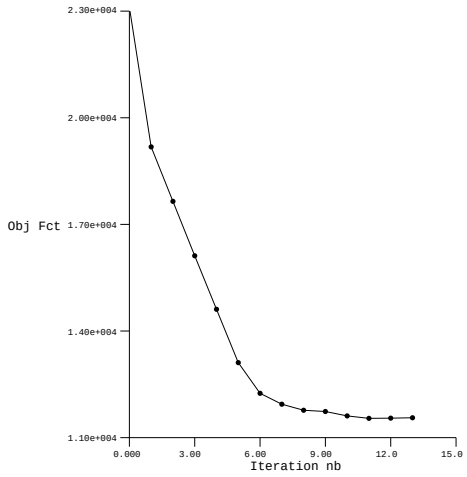
Figure 6.20: Initial geometry and analysis domain with parametrization

In Figure 6.20 (a), the analysis domain (i.e. meshed domain) is represented with dashed line. The 3 circles $C_1, C_2, C_3$ are discretized with a conforming mesh for an easy application of the essential boundary conditions. These circles are the unique geometrical features explicitly modeled, the outer boundary being modeled implicitly with a complex Level Set object.

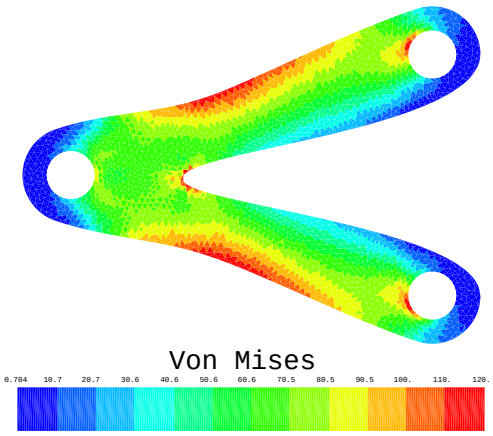
The analysis of the initial geometry given in Fig. 6.20 (b) clearly shows that a lot of mass can be saved. Indeed, most of the material is far from reaching the limit value of 120 $N/mm^2$ and the highest stresses are localized around the two fixations.
The optimal shape illustrated in Figure 6.21 (b) is obtained after 13 iterations. A reduction of 48 % of the intial volume is realized. The maximum Von Mises stress is now located on the boundary $h_1$ with a maximum value of 120 $N/mm^2$ as prescribed. As expected, the geometry is symmetric with respect to an horizontal axis as the two arms joining $C_1$ to $C_2$ and $C_1$ to $C_3$ are subject to the same loading, the lower arm being in compression and the upper arm in traction.

### Complete geometry

Now, the hole $h_1$ is introduced and modeled as a complex Level Set composed of 3 circle arcs and 3 line-segments. First, the previous optimization problem is replayed considering exactly the same design variables and relative side constraints. The hole $h_1$ remaining constant and fixed. After 13 iterations, an optimal solution is reached (see Fig. 6.22 (b)). In this solution, the structure experiences a modification of its initial topology as the initial hole $h_1$ has completely



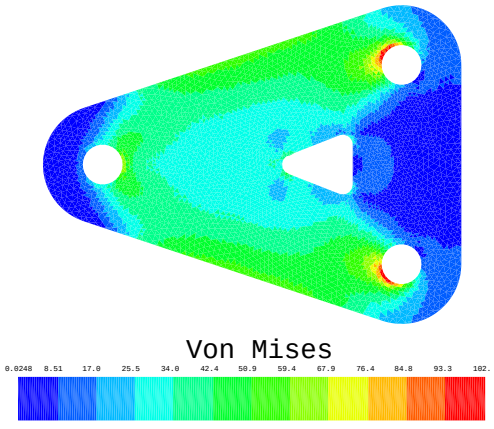

(a) Objective function history



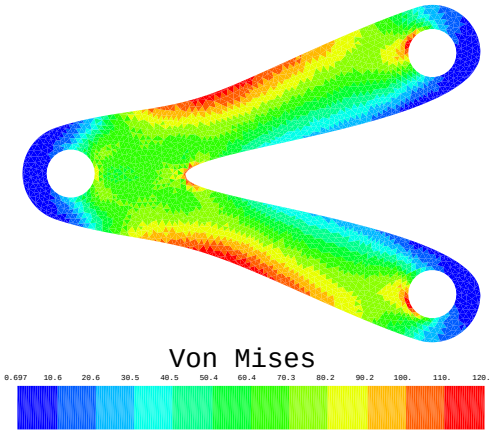

(b) Optimal shape with Von Mises distribution ($N/mm^2$)

Figure 6.21: Optimization history and final shape

disappeared. The removal of this entity is possible by the movement of the external boundary up to a position completely overlapping the Level Set representing $h_1$. Moreover, this solution is very close to the solution found in the first problem. The difference in the value of the final parameters are less than 1 %.




(a) Initial shape with Von Mises distribution ($N/mm^2$)

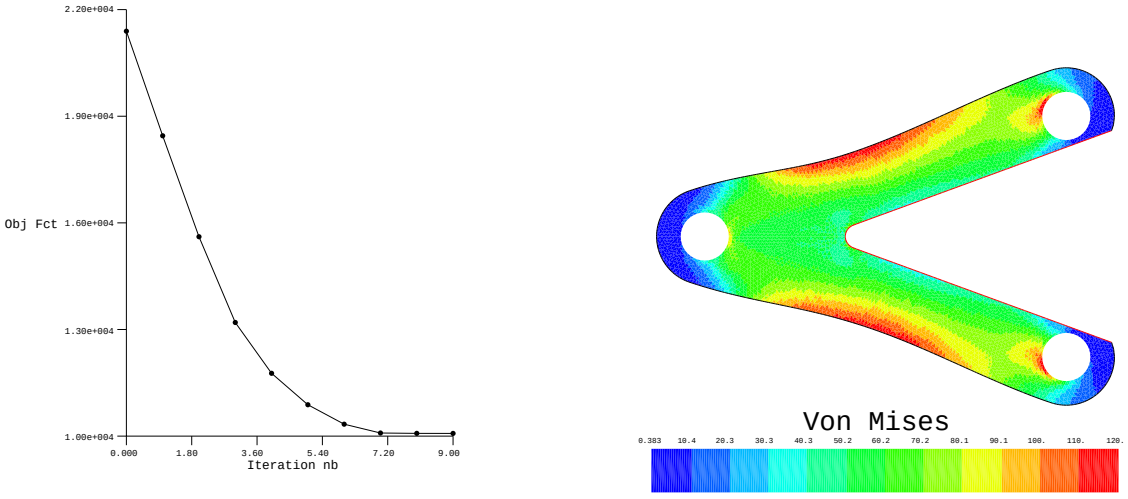



(b) Optimal shape with Von Mises distribution ($N/mm^2$)

Figure 6.22: Initial geometry of the 2D bracket support

Taking into account the design of the inner hole geometry as an additional design variable of the problem, the rounded triangle $h_1$ is parametrized using 3 independent variables (see Fig. 6.18).

The first variable is the position of the left circle. The second variable is the distance along the $y$ direction between the left circle and the upper right circle. The third variable is the height of the upper right circle of the rounded triangle. The two right circles being constrained to be symmetric with respect to the $y$ axis. The results of the novel optimization is represent in Figure 6.23.




(a) Optimization history

(b) Optimal shape with Von Mises distribution ($N/mm^2$)

Figure 6.23: Optimal shape and convergence history considering both inner and outer boundaries as variables

In this optimal geometry (see Fig 6.23 (b)), the introduction of the rounded triangle as a variable of the problem allows to obtain a lighter design as the mass reduction is now equal to 54% compared to 48 % for the first problem. The gain of performance is due to the introduction of the inner Level Set that enables the optimizer to reduce the volume where low-stresses are present. Hence, we can remark that the boundary of the structure is now divided in two parts. The first part is represented by the initial Level Set (black contour in Figure 6.23 (b)) that models the external boundary and the inner boundary that is represented by the second Level Set (red contour in Figure 6.23 (b)). Treating the same problem with a FEM approach would be much more difficult as a special treatment would be necessary at some stage of the process to detect the collisions and the degeneracy between geometric features and to redefine the new boundaries of the structure geometry.

The Figure 6.23 (a) representing the evolution of the objective function shows that the optimization process evolves progressively and smoothly to the optimum without any oscillations. However, we should point out that the procedure is sensitive to the mesh adopted for the analysis. Depending on the refinement of the mesh the optimizer may stop prematurely.

### 6.4.2 Compliance minimization with volume constraint

Here, the problem is formulated as a classical topology optimization one. We seek to minimize the compliance with a target value on the mass of the structure. The geometrical model is also modified as we focus only on the shape of the inner hole, keeping the external boundary

unmodified. The mesh (see Fig. 6.24) is built using elements conforming to the external boundary and the characteristic size of the elements is higher compared to previous problem as we treat a compliance problem that does not need a high accuracy at the element level. The inner triangular hole is still modeled with a rounded triangle and the design parameters are the same than in previous section. By inspecting the solution obtained by Duysinx [57], we know that the hole should be ideally positioned at the right extremity of the structure. Therefore, in order to start with a solution not too close to the optimal solution we modify the initial position of the hole to the left using the following values: $t$=30 mm, $s$=65 mm, $k$=10 mm.

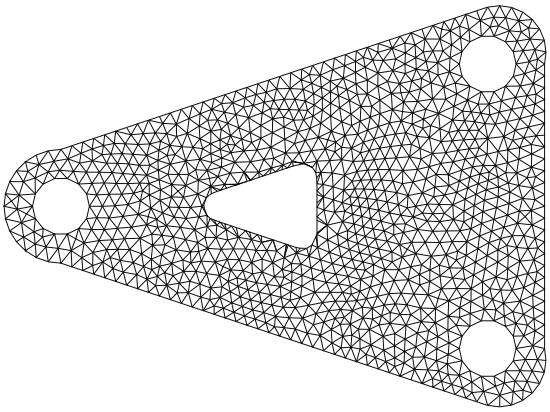
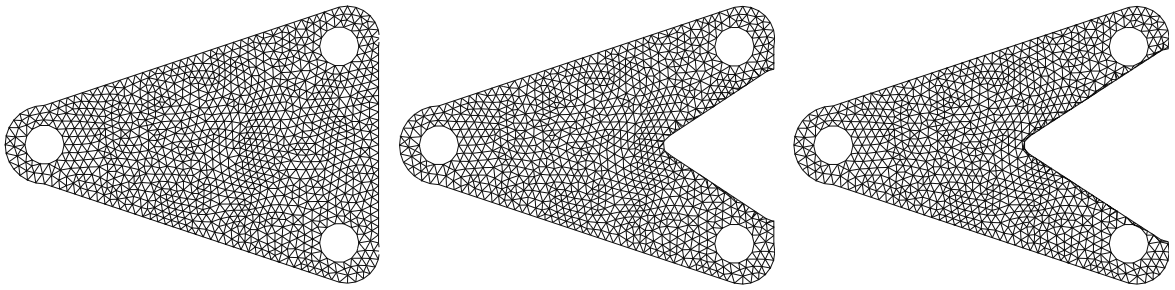


Figure 6.24: Initial geometry and mesh discretization

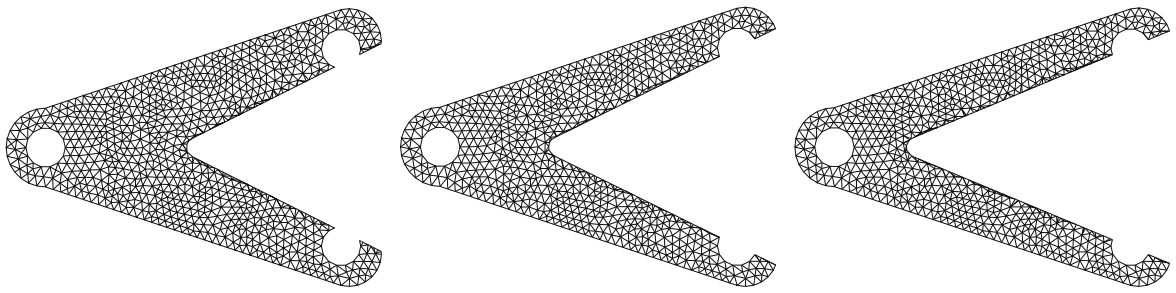
The optimization process is repeated several times with different bounds on the volume ($V$) constraint: $V \leq 1.5V_i$, $V \leq 0.9V_i$, $V \leq 0.8V_i$, $V \leq 0.6V_i$, $V \leq 0.7V_i$, $V \leq 0.5V_i$ where $V_i$ is the initial volume of the structure. The Figure 6.25 illustrates the results obtained for the several bounds. In Figure 6.25 (a), corresponding to the constraint $V \leq 1.5V_i$, the optimizer takes advantage of the large volume resource to remove completely the hole in order to minimize the compliance of the structure. This topological change is not impressive but it shows that it is possible to remove a geometrical entity which is impossible with a classical FEM model unless the case is explicitly treated in the geometrical modeler.



(a) $V \leq 1.5V_i$ (b) $V \leq 0.9V_i$ (c) $V \leq 0.8V_i$

Figure 6.25: Optimal solutions for different bounds on the volume constraint

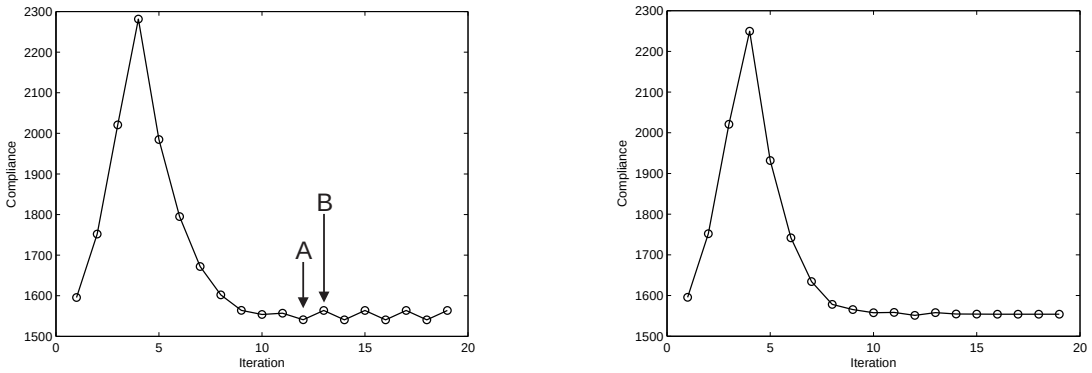
For the cases (b) and (c), the volume constraint does not allow to remove the hole as the final volume should be lower than its initial value. But we can observe that the optimizer moves the hole towards the right side, proceeding to a modification of the topology, since the hole is opened.



(a) $V \leq 0.7V_i$ (b) $V \leq 0.6V_i$ (c) $V \leq 0.5V_i$

Figure 6.26: Optimal solutions for different bounds on the volume constraint

Further reducing the admissible quantity of material for the structure constrains the optimizer to increase the size of the hole. Comparing these solutions with the optimal density distribution obtained in Duysinx [57], we can remark a common trend between the two approaches making the arms to reduce the compliance. Here, the optimizer prefers to increase the height of the triangle rather than its width even if a part of the fixation is removed to obtain a larger moment of inertia.

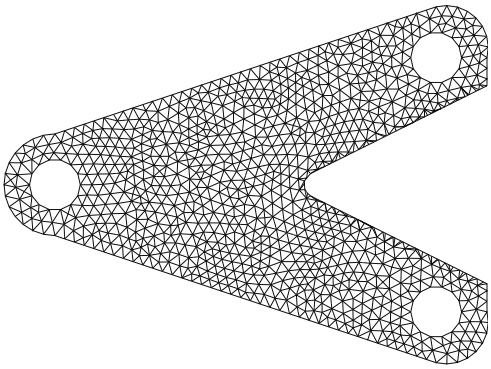



(a) Convergence history without move limits strategy (b) Convergence history with move limits strategy

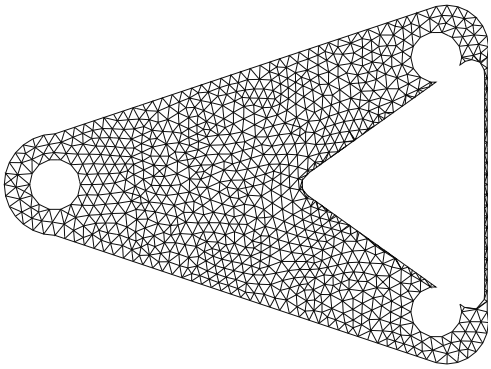
Figure 6.27: Convergence history for the problem with constraint $V \leq 0.8V_i$

All these problems, except those with highest bounds on the volume, suffer from convergence oscillations when the current design gets close to the optimal solution. The situation is illustrated in Figure 6.27 (a). The solution starts oscillating from the iteration 11. In fact, these oscillations appear between two configurations depicted in Figure 6.28 (a) and (b) which correspond to the point where there is a modification of the topology. Hence, when the optimizer removes the part joining the circles $C_2$ and $C_3$, it effectively reduces the compliance but the volume increases slightly and violates the constraint. Then, the optimizer steps back to the previous design. In order to prevent these oscillations and to obtain a smooth convergence process as illustrated in Figure 6.27 (b), it is necessary to introduce an adaptive move limit strategy.

Imposing a tight constraint on the volume can lead to ill-defined geometries (see Fig. 6.29 (a)).



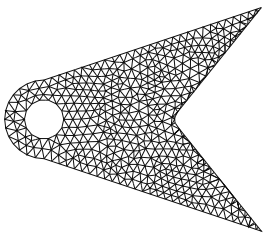
(a) Geometry of iteration $A$ of Figure 6.27 (a)



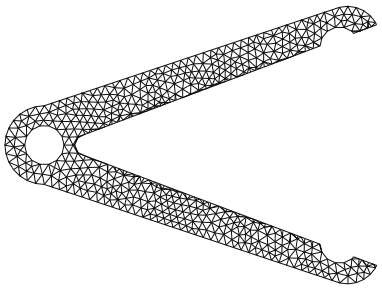
(b) Geometry of iteration $B$ of Figure 6.27 (a)

Figure 6.28: Geometries of iterations $A$ and $B$ of Figure 6.27 (a)

Indeed, at first, the optimizer tries to reduce the constraint violation and find a feasible design. To this end, the size of the hole is increased and it is possible to obtain a structure split in two parts leading to a degenerated geometry with rigid modes and infinite compliance. For instance, this particular problem occurs if we impose a volume constraint $V \leq 0.4V_i$. A degenerated geometry represented in Figure 6.29 (a) is obtained after 4 iterations. The fixations being completely included in the negative part of the Level Set, the structure does not have any fixations left and there are several rigid modes. More dramatically, the sensitivities on the compliance become null and the optimizer is stuck at this design without any possibilities to back track. To circumvent the problem, two strategies are available. The first one is to define geometrical constraints. In practice, it is cumbersome and it can be very difficult to define geometrical constraints that do not reduce the design domain. Another strategy that can be implemented consists in adding a soft material to the void domain. Hence, even if the optimizer reach an ill-defined geometry, it always has the possibility to recover the design to a feasible one. Indeed, adopting this approach for the previous design and $V \leq 0.4V_i$ enables the optimizer to reach an acceptable solution shown in Fig. 6.29 (b).



(a) Evolution stopped at iteration 4 due to rigid mode detection



(b) Optimal solution with added soft material in the void

Figure 6.29: Optimal solutions for volume constrain $V \leq 0.4V_i$

**Robustness and accuracy of the sensitivity analysis**

If we use the '*ignore strategy*' to compute the sensitivities in the previous compliance minimization examples, we face a single model in which there is creation of DOFs during the geometrical perturbation. This illustrates that in most cases, the simplest strategy is generally sufficient to obtain accurate sensitivities while pathologic configurations occur rarely.

To illustrate a case where new DOFs are created, let's consider the following geometry shown in Figure 6.30 and minimize the compliance considering the external boundary parameterized with 2 NURBS control points on each curve. Contrary to the previous models, the load is aligned along the $x$ axis in order to study a symmetric problem. The control points 1,2,5,6 move along the $y$ axis and the points 3,4 along the $x$ axis.

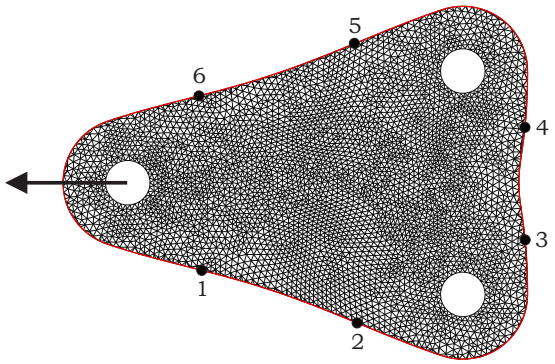



Figure 6.30: Geometry and variable perturbation

With this configuration, we compute the sensitivity of the compliance considering different perturbations and the results are given in Table 6.2.

| $\delta$x | El. | x | $\Delta$x | $dC/dx$ | $dof$ | Rel. dif. % | x | $\Delta$x | $dC/dx$ | $dof$ | Rel. dif. % |
|---|---|---|---|---|---|---|---|---|---|---|---|
| -1E-4 | 460 | $P_1$ | -0.00598 | 0.2039812 | 0 | – | $P_6$ | -0.01401 | -0.2040258 | 0 | – |
| -1E-3 | 460 | $P_1$ | -0.05987 | 0.2039899 | 2 | 4E-3 | $P_6$ | -0.14012 | -0.2040144 | 0 | – |
| 1E-4 | 460 | $P_1$ | 0.00598 | 0.2039782 | 0 | – | $P_6$ | 0.01401 | -0.2040283 | 0 | – |
| 1E-3 | 460 | $P_1$ | 0.05987 | 0.2039647 | 0 | – | $P_6$ | 0.14012 | -0.2040439 | 2 | 8E-3 |
| -1E-4 | 460 | $P_2$ | -0.00432 | 0.1733148 | 0 | – | $P_5$ | -0.01567 | -0.1732840 | 0 | – |
| -1E-3 | 460 | $P_2$ | -0.04320 | 0.1732760 | 2 | 2E-2 | $P_5$ | -0.15679 | -0.1732656 | 0 | – |
| 1E-4 | 460 | $P_2$ | 0.00432 | 0.1733108 | 0 | – | $P_5$ | 0.01567 | -0.1732880 | 0 | – |
| 1E-3 | 460 | $P_2$ | 0.04320 | 0.1732934 | 0 | – | $P_5$ | 0.15679 | -0.1733050 | 0 | – |
| 1E-3 | 460 | $P_3$ | 0.40500 | -0.0202156 | 2 | 1 | $P_4$ | 0.40500 | -0.0201340 | 2 | 0.9 |
| 1E-4 | 460 | $P_3$ | 0.04050 | -0.0199604 | 0 | – | $P_4$ | 0.04050 | -0.0200002 | 0 | – |
| -1E-4 | 460 | $P_3$ | -0.04050 | -0.0198990 | 0 | – | $P_4$ | -0.04050 | -0.0199373 | 0 | – |
| -1E-3 | 460 | $P_3$ | -0.40500 | -0.0196260 | 0 | – | $P_4$ | -0.40500 | -0.0196527 | 0 | – |

Table 6.2: Comparison of sensitivities for different perturbation steps

In this table, $\delta x$ corresponds to the relative perturbation step and $\Delta x$ is the finite perturbation step. The column 'El.' indicates the number of elements on which the sensitivity is computed and '$dof$' is the number of DOFs created during the perturbation phase. Table 6.2 shows that new DOFs are created for the perturbation of all variables except $P_5$. In each case, the number of DOFs created is rather small and the error becomes negligible. Furthermore, we can also

notice that the sensitivity of variable $P_1$ and $P_6$ should be equal in absolute value as the model is symmetric, which is confirmed in the results. The same remark can be stated for the points $P_3$ and $P_4$.

## 6.5 2D suspension arm

The suspension arm problem has been initially proposed by Bennet and Botkin [30] and later investigated amongst others in Braibant [37], Zhang [183] and also Duysinx *et al.* [56]. The goal of the optimization problem is to minimize the weight of the arm represented in Figure 6.31 by modifying the external and the internal boundaries while keeping the Von Mises stresses below $\sigma_{eq}$=80E3 N/cm$^2$.

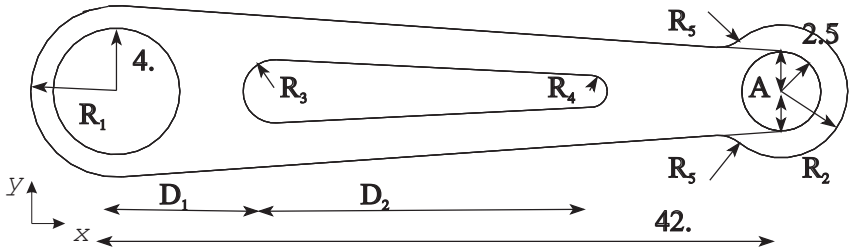



Figure 6.31: Geometry of the suspension arm

The initial values of the geometrical parameters are: $R_1$=7 cm, $R_2$=7 cm, $R_3$=1.5 cm, $R_4$=1 cm, $D1$=12 cm and $D_2$=15 cm. The structure has a length of 50 cm, a maximum width of 14 cm and the thickness is 3 mm. The load case is composed of $F_x$=2789 N and $F_y$=5066 N. Both loads acts in the plane XY and are uniformly distributed on the right circle whereas the left circle is clamped. The properties of the material are: Young modulus $E$= 2.074E6 N/cm$^2$, Poisson's ratio $\nu$=0.3 and density $\rho$=7.81E-3 kg/cm$^3$.

The design domain has a rectangular dimension of 70 cm × 20 cm meshed with approximately 7000 linear triangles. Only the two internal circles are represented with conforming elements to ease the application of the boundary conditions. To model the inner and outer boundaries, we use complex Level Sets composed of a set of different parametric geometrical objects. In all test cases, only the upper part (positive $y$ positions) of the boundary is parametrized, the lower part (negative $y$ positions) being constrained to be symmetric with respect to the $x$ axis.

### 6.5.1 Geometry without central hole

Imagine that in a preliminary analysis the engineer investigates a very simple design and does not insert a central hole in the initial structure model. Given the previous solutions obtained in the literature [30, 37, 183], the outer boundary is modeled with a Level Set that is able to describe the geometrical boundary of the external contour as in Figure 6.31. So, this Level Set contains two circular arcs, one piece of line and one NURBS curve. In Figure 6.32, from left to right, the first circle arc joins $x$ axis (Point $P_1$) and point $P_2$, the line joins points $P_2$ to $P_3$, the NURBS curve joins points $P_3$ to $P_4$ and the second circle arc joins the point $P_4$ to the $x$ axis (Point $P_5$). With this Level Set description, we have a total of 3 variables: the two circle

radii ($R_1$ and $R_2$) and the $y$ position of the point $P_3$. The angular sector of the two circle arcs remain fixed at 90° and the NURBS curves control points are all determined with the imposition of tangency conditions between the line $P_2$-$P_3$ and the second circle arc $P_4$-$P_5$.

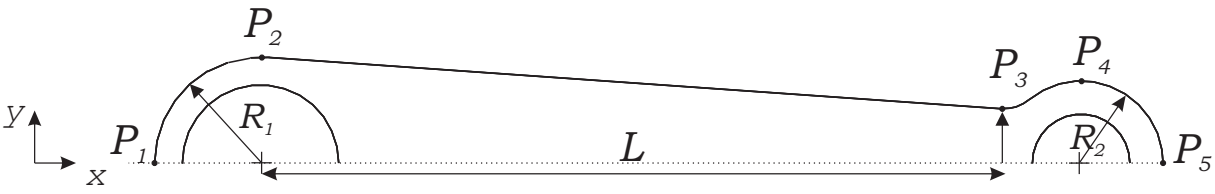
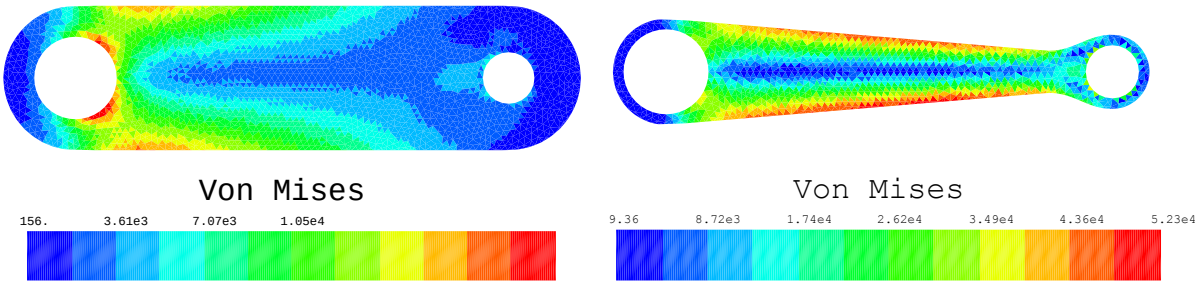



Figure 6.32: Parametrization of the exterior contour

The initial values of the parameters are: $R_1$=7 cm, $R_2$=7 cm and $P_{3y}$=7 cm, $L$= 0.85 ($P_{4x}$-$P_{2x}$) and the minimal admissible values are $R_1 \geq 5$ cm, $R_2 \geq 3.5$ cm $P_{3y} \geq 1$ or 2 cm. The initial weight is 5.23 kg and the maximum Von Mises stress is 20.9E3 N/cm$^2$ (see Fig. 6.33 (a)). The CONLIN optimizer is used.




(a) Initial Von Mises stresses distribution

(b) Von Mises stresses at final iteration

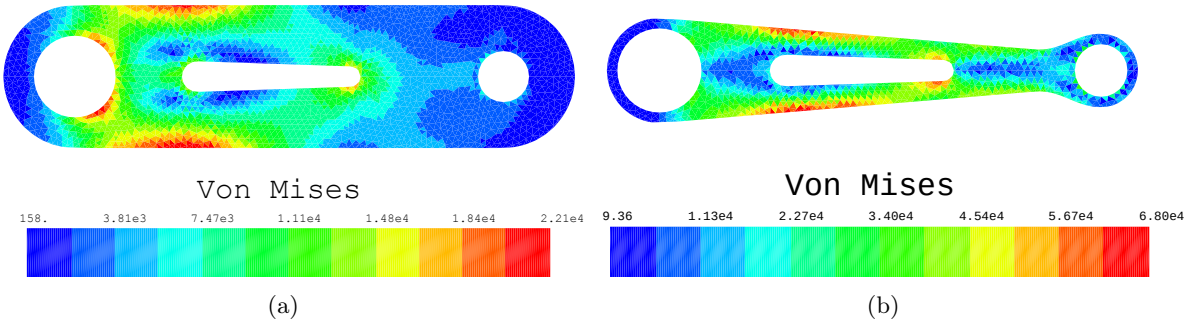
Figure 6.33: Optimal solution for the first parametrization

The optimal shape presented in Figure 6.33 (b) is obtained after 14 iterations. The design variables have the following values: $R_1^*$=5 cm, $R_2^*$=3.5 cm and $P_{3y}^*$=1.44 cm. For this design, an important reduction of the weight is achieved. The mass is now equal to 1.91 kg, corresponding to a gain of 267 %. Inspecting the values of the design variables, we can see that the two radii reach their minimal values but the Figure 6.33 (b) suggests that, without taking into account manufacturing considerations, some mass reduction could be further obtained as the Von Mises stress around the two circles is far from the limit value $\sigma_{eq}$. Moreover, we can observe that the variable representing the $y$ position of the point $P_3$ has an important influence on the Von Mises stress. For instance, if we assign the minimal value of 2 cm for $P_{3y}$, the maximum Von Mises stress remains under the prescribed value $\sigma_{eq}$.

### 6.5.2 Geometry with central hole

To lower the mass of the suspension arm, one can introduce a central pocket as suggested in the initial design proposed by Bennet and Botkin [30] (see Figure 6.31). Thanks to the fixed mesh approach and the Level Set geometrical representation this can be achieved directly using the previous model with minor modifications. We only need to insert a second Level Set

representing the central hole and combine it (using Level Set Union boolean operator) with the initial Level Set representing the outer contour as depicted in Figure 6.34 (a). In this example, GMSH needs 0.3 s to generate the mesh (15900 triangles) of the analysis domain and OOFELIE takes 0.032 s to slice the mesh (1020 elements are cut). Replaying the previous optimization process with exactly the same set of design variables and constraints, one gets the solution given in Figure 6.34 (b).




(a) (b)

Figure 6.34: Initial and Final Von Mises distribution

This optimized design has a lower mass compared to the previous optimized geometry as the introduction of the central hole enables to obtain an extra mass reduction of 300 g. The final mass is equal to 1.6 kg by comparison to 1.9 kg without the hole. The gain is equal to 300 % with respect to the intial design of Figure 6.34 (a) and 400 % with respect to the intial design without the central hole. For this configuration, the design variables have the following final values: $R_1$=5 cm, $R_2$=3.5 cm and $P_{3y}$=1.44 cm. These values are exactly the same as in the previous optimization problem, which indicates that a better performance can be achieved by modifiying the central hole.

### Optimizing internal and external boundaries

The performance in terms of mass reduction obtained for the two previous models are important but it can be further improved by changing the shape of the central hole. Parameterizing the central hole with 4 independent variables: $R_3$, $R_4$, $D_1$ and $D_2$ representing the radii and the position of the two circular arcs ($P_5$-$P_6$ and $P_7$-$P_8$) as illustrated in Figure 6.35, it is possible to modify the geometry of the internal boundary of the suspension arm.

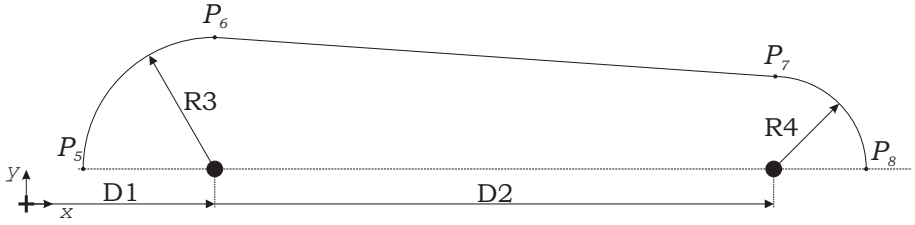
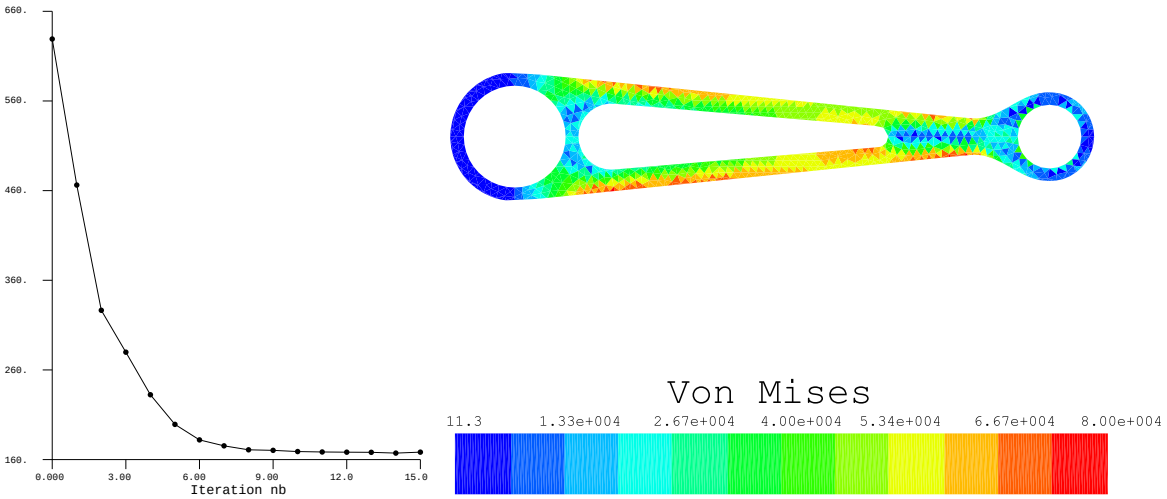



Figure 6.35: Parametrization of the internal contour

Introducing these new variables into the optimization problem, with the following side con-

straints: $0.5 \leq R_4 \leq 4$, $1.0 \leq R_3 \leq 4$, $6 \leq D_1 \leq 20$ , $5 \leq D_2 \leq 32$, we obtain the solution represented in Figure 6.36 (b) after 14 iterations.




(a) History of the objective function

(b) Optimal design

Figure 6.36: Optimal connecting rod geometry with variable external and internal boundaries

Compared to the initial design, the introduction of the internal hole allows a larger mass reduction of 377 % to 1.3 kg. Moreover, we can observe that the introduction of the 4 additional design variables into the optimization problem does not increase the number of iterations to reach an optimal solution. In Figure 6.36 (a), representing the objective function history, we can observe that the first iteration provides an important mass reduction while in the following iterations the mass is reduced with a slower and decreasing rate. This can be explained by the fact the optimizer modifies strongly the values of the two external radii because these design variables allow an important mass reduction in a zone where the Von Mises stress is far from the prescribed limit value $\sigma_{eq}$. After two iterations, the radii touch their lower bounds and CONLIN optimizer has no other choice than modifying the other variables to achieve a better design. Comparing our geometry with the solution obtained by Duysinx *et al.* [56] (represented in Figure 6.37 top), we can see that both designs are very similar. Two minor differences can be noticed: the fillet between the arms and the right external circle are not totally identical and the arms are slighty thicker for the FEM solution. The first difference can be related to the fact that the geometrical parametrization used for the FEM and the X-FEM model are not exactly the same. The second difference is due to the introduction of a minimal thickness constraint of 1 cm in FEM while there is no such limitation for the X-FEM model. However, we can see that the optimal arm thickness obtained with X-FEM is very close from the FEM solution. Indeed, the Von Mises restriction prevent too thin arms designs. Finally, to obtain the design of Figure 6.37 we have to introduce a restriction between the outer and the inner boundaries to prevent the left circle arc $P_5$-$P_6$ of the internal hole to overlap the left circle where the fixations are applied.

For the first iterations of the analysis, the initial mesh presents an adequate element density and is rather fine regarding the considered geometry. However, when the external and the internal Level Set boundaries get closer to each other, the two arms being thinner, and the mesh becomes quite coarse in these zones as well as near the left and right circles. The fixed mesh approach is

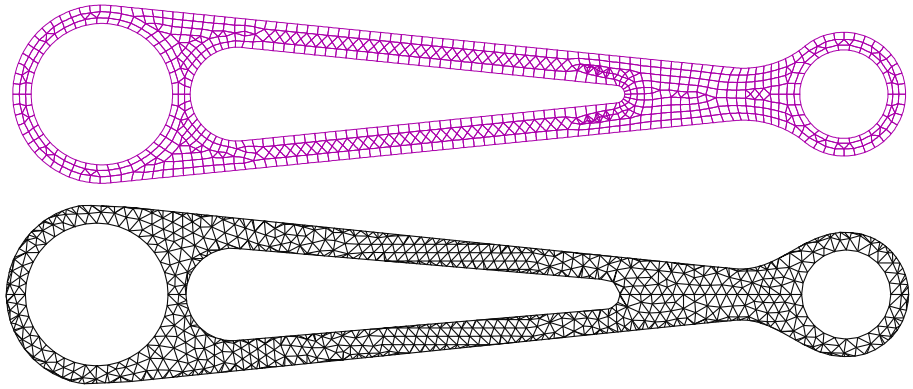


Figure 6.37: Comparison of FEM [56] and X-FEM solution

in this case very useful as the initial geometry experiences a large modification during the optimization but it also presents a serious drawback as the element density should be monitored and adapted to maintain an acceptable accuracy as long as the shape gets thinner. This problem is not specific to fixed grid methods and has already been pointed out in shape optimization using FEM. Duysinx *et al.* [56] have shown that the introduction of an error control strategy and adaptive mesh refinement within the optimization loop has an influence on the shape evolution and leads to a better optimal geometry.

Unfortunately, up to now, no research on error estimator and mesh adaptation techniques has been reported for the X-FEM and specifically for the material-void approach. We can only mention the work by Bordas, Duflot and Le [35, 54] that focuses on the introduction of an *a posteriori* error estimator in X-FEM for the analysis of structures presenting cracks.

Henceforth, without an error indicator to define where the mesh should be refined, we propose to refine the mesh with an indicator based only on the geometry of the structure. As previously shown in the chapter 2, Moës *et al.* [113] presented a simple mesh refinement criterion based on the Level Set values in order to obtain an accurate geometric representation of the iso-zero Level Set. However, in our case, this idea is not very relevant here as the problem of the mesh refinement is related to the proximity of several Level Sets to each others as well as the distance between the Level Sets and the analysis domain boundaries. Therefore, we introduce a simple strategy that consists in refining the mesh at the proximity of the structural boundary. To this end, the Level Set description is again very handful as it can provide immediately a useful information on the boundary location and on the distance to the domain boundaries which can be used to define a characteristic size for the mesh.

Reminding that the Level Set is constructed as a signed distance function, the absolute value of Level Set (see Figure 6.38) reaches a minimum value close to the boundary which can be chosen as characteristic size $l_c$ for the mesh. Indeed, low Level Set values close to the boundary will provide a dense mesh at the vicinity of the structure boundaries whereas the mesh size will be progressively enlarged elsewhere. Practically, using the absolute value of the Level Set function 'as is' to define the characteristic size is not very appropriate because the Level Set function presents a too high variation leading to a mesh that varies quickly from very small element mesh size to coarse ones. Moreover, if a maximum and minimum $l_c$ value is not defined, the lowest

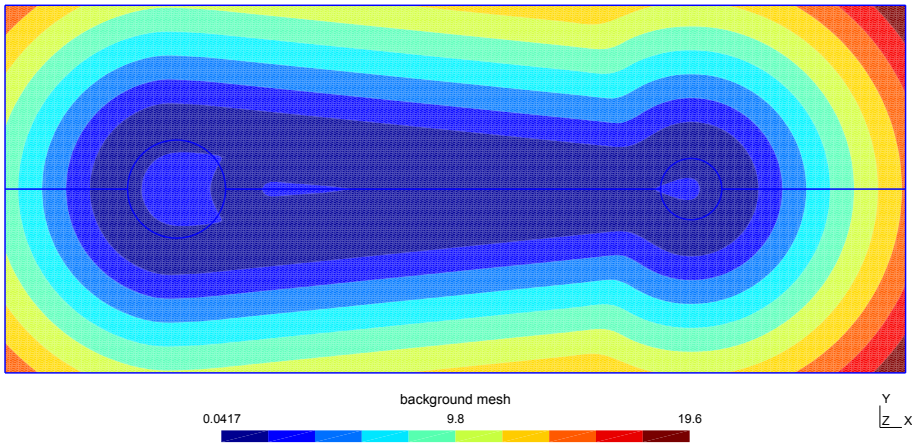



Figure 6.38: Absolute value of the Level Set function for the suspension arm

(highest) $l_c$ values will generate too small (big) elements. To obtain a smoother variation, we compute a size mapping by defining a characteristic element mesh size $l_c$:

$$l_c = \alpha \, |ls_i|^{\beta} \tag{6.3}$$

where $\alpha < 1$ is a factor that controls the overall refinement, $0.5 \leq \beta < 1$ controls the range of variation of the mesh size and $ls_i$ is the nodal Level Set value. Using (6.3), we obtain the size map illustrated in Figure 6.39 where we can see that the variation of $l_c$ is smoother ($l_c$ ranging between 0.2 and 1.42 whereas using the absolute Level Set values, this mesh size would range from 0.0417 to 19.6).

In Figure 6.39 (b), we can see that the size map provides an acceptable mesh which is refined only at the neighborhood of the boundary of the structure and enlarges progressively towards the analysis domain boundaries.

Following the same idea, obtaining a constant mesh size over the whole structural domain or in a prescribed band width around the boundary can be directly achieved by imposing a mesh size $l_c$ where the Level Set value is lower than a given threshold $\bar{ls}$. Doing so, we obtain a size map that is completely constant and equal to the prescribed mesh size inside the structure as illustrated in Figure 6.40 (a). Using this size map for generating the mesh, we obtain the mesh shown in Figure 6.40 (b) that presents now a constant mesh size on all the structural domain. Finally note that to obtain the presented size map, the Level Set values are not computed on the structural mesh but on finer Cartesian background grid in order to obtain a higher accuracy on the Level Set values, to speed up the Level Set generation and proceed to proceed to a Laplacian smoothing to remove the peak values for nodes close to the iso-zero boundary.

Incorporating the boundary mesh refinement strategy for the shape optimization of the suspension arm after the first iteration, we obtain the optimal shape represented in Figure 6.41.

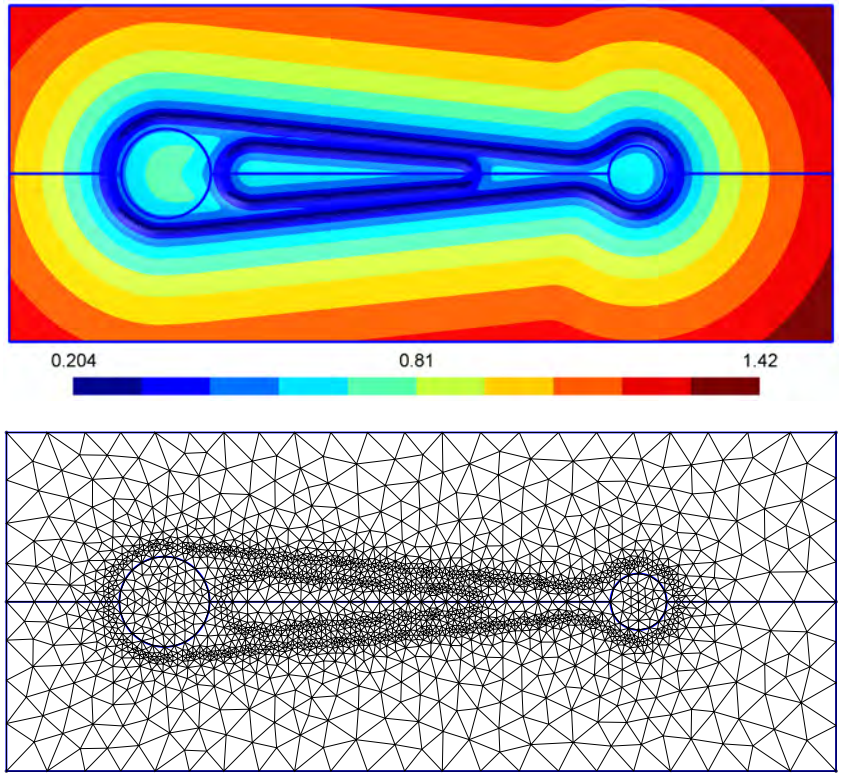



Figure 6.39: Size map and mesh with $\alpha = 1$ and $\beta = 0.5$

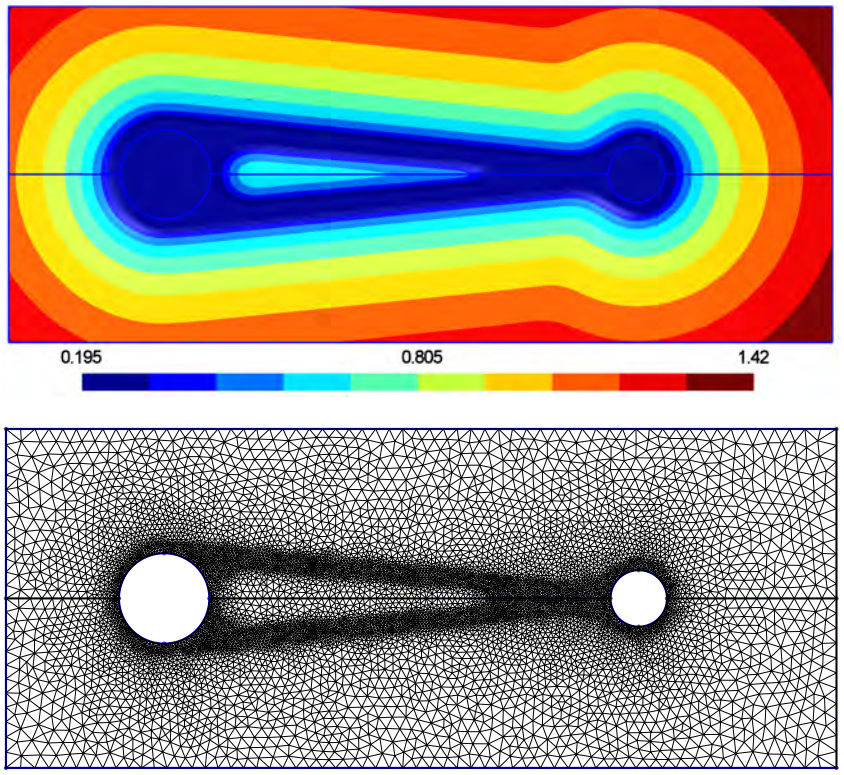



Figure 6.40: Size map and mesh with $\bar{ls}$=0.1

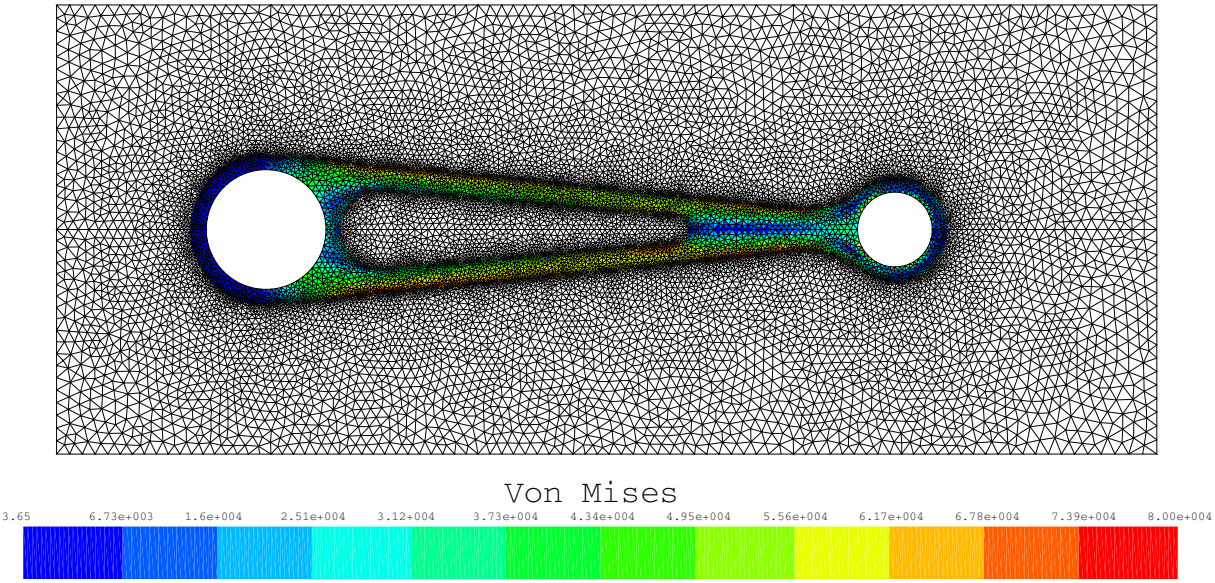



Figure 6.41: Optimal suspension arm design with boundary adapted mesh

Compared to the solution without mesh refinement, we observe the same trend than observed by Duysinx [56]. The reduction rate of the objective function between iterations is lower and we obtain a solution that presents a higher mass (Fig. 6.42). The most important difference between the standard mesh and the adapted mesh solutions concerns the central hole which is smaller in the adapted case. This result is quite natural as the finer mesh allows to compute more accurate stress values, which influences the optimization process as the higher stresses (located on the central hole boundary) are under estimated with the coarse mesh. The Table 6.3 summarizes and compares the different solutions. We can see that using the adaptive mesh strategy, the solutions are slightly heavier than without refining the mesh.

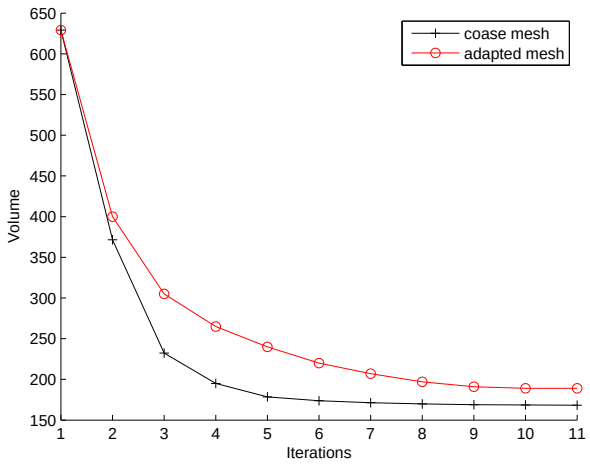



Figure 6.42: Comparison of the history of the objective for the coarse and adapted mesh model

| Geometry configuration | Initial mass (kg) | Final mass (kg) | Mass gain (%) |
|---|---|---|---|
| No central hole | 5.23 | 1.91 | 273 |
| Constant central hole geometry | 4.91 | 1.6 | 300 |
| Variable central hole geometry | 4.91 | 1.3 | 377 |
| Variable central hole geometry with adapted mesh | 4.91 | 1.42 | 340 |

Table 6.3: Comparison of performance of the different geometrical models

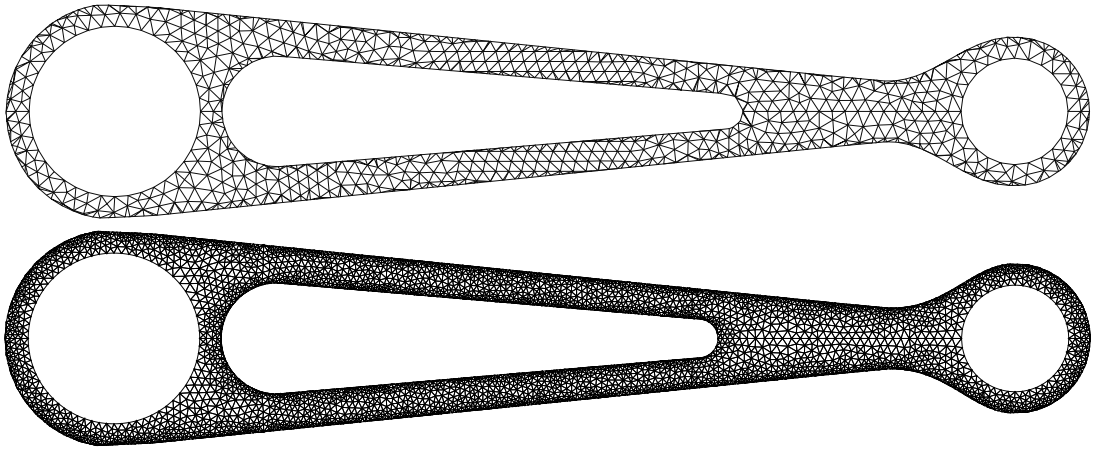


Figure 6.43: X-FEM coarse and adapted mesh optimal solutions

## 6.6 Shape optimization of a 2D dam

This test case illustrates the shape optimization of a structure with design-dependent loads. We focus on a well known structure that exhibits this type of loading: a water dam. In the complete design process of a dam, several types of loadings should be taken into account such as the static loading due to the water pressure, earth-quake loadings, waves, elasticity of the foundations, porosity and seepage. However, in the present study, we consider a simplified load case that is the hydrostatic pressure of the water. A similar problem has been treated in [72] considering compliance minimization with a topology optimization method.

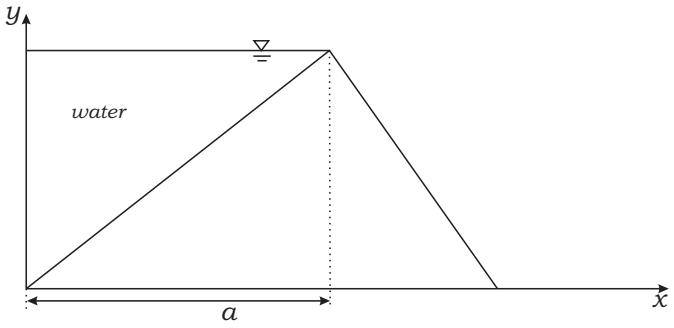



Figure 6.44: Shape parameter design for parametric study

Intuitively, the optimal cross section of the dam should be close to a triangle shaped structure. To obtain a preliminary design shape that can serve as a reference performance value, a parametric

study is realized by changing the position of the apex $a$ of the triangle represented in Figure 6.44.

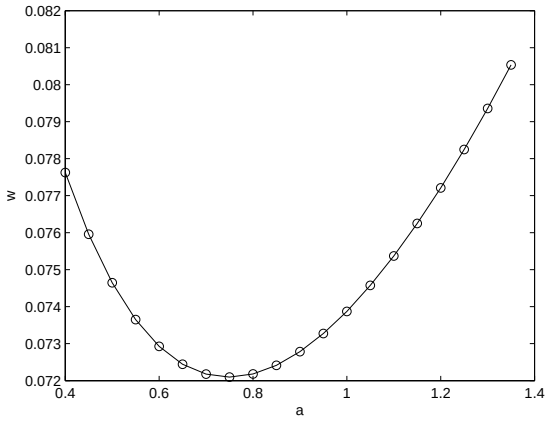



Figure 6.45: Variation of the strain energy with respect to design variable $a$

The Young Modulus is normalized at $E$=1 N/m$^2$, the Poisson ratio $\nu$=0.2 according to [72] and a plane strain state is assumed. The dam foundations are clamped and have a length $l$=2 m. The height is $h$=1 m, the density of the water is 1E3 kg/m$^3$ and the gravity $g$ is equal to 1 m/s$^2$. Figure 6.45 plots the corresponding strain energy for varying shapes of the triangular dam. We can see that the optimal position of the apex is $a \simeq$0.75 m and the strain energy is equal to 0.0720 J.

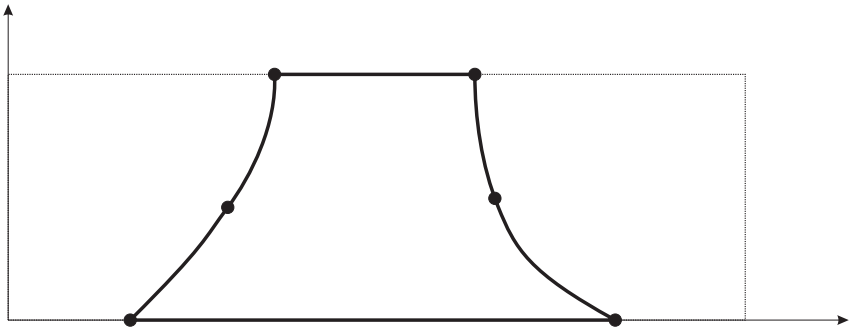


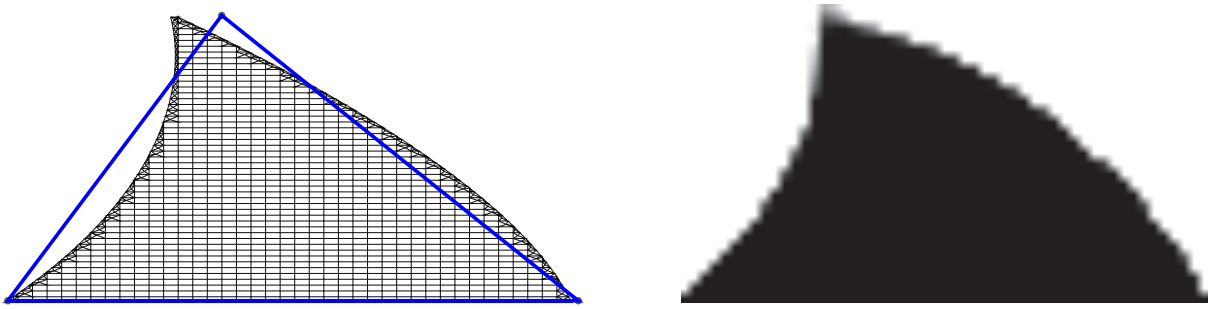
Figure 6.46: Shape parameter design and design domain for optimization problem

For the optimization, we consider a rectangular design domain of dimension 3 m $\times$ 1 m as illustrated in Figure 6.46. Both upstream and downstream surfaces are modeled using NURBS curves with 3 control points placed at $y$=0, 0.5, 1.0 m. The upstream surface is subject to the hydrostatic pressure while the downstream surface is free of traction. The initial position of the downstream surface is located at $x$=2.5 m and the upstream surface is placed at $x$=0.5. The optimization variables are the position of these control points along the $x$ axis. However, to compare the optimized geometry with the parametric analysis, the length of the foundations is fixed to 2 m. The optimization formulation aims at minimizing the compliance with a prescribed amount of material $V \leq 1$ m$^3$. The hydrostatic pressure $\rho g h$ is applied on the upstream iso-zero Level Set and acts perpendicularly to the structure boundary. Notice that this example also

illustrates that to apply the loading, it is necessary to tag each boundary intersection of the Level Sets with the mesh in order to select the appropriate surface.

In this application, the loads depend on the shape of the boundary in contact with the water. Therefore, we have to take into account of the variation of loads when computing the sensitivity of the compliance $C$ as mentioned in chapter 5.6 using:

$$\begin{aligned} \frac{\partial C}{\partial z} &= \mathbf{u}^T \frac{\partial \mathbf{K}}{\partial z}\mathbf{u} + 2\mathbf{u}^T \mathbf{K}\frac{\partial \mathbf{u}}{\partial z} \\ &= -\mathbf{u}^T \frac{\partial \mathbf{K}}{\partial z}\mathbf{u} + 2\mathbf{u}^T \frac{\partial \mathbf{f}}{\partial z} \end{aligned}$$



(a) Optimal X-FEM geometry with optimal triangular dam

(b) Solution obtained by Fuchs and Shemesh [72]

Figure 6.47: Optimal solution with a volume constrain $V \leq 1$ m$^3$

The optimal solution is obtained after 11 iterations and is illustrated in Figure 6.47 (a). We can see that the result obtained with the parametric analysis is very close to the solution obtained with our shape optimization. The optimal geometry has a strain energy $W$=0.068 J that corresponds to a gain of 5 % with respect to the optimal triangular dam. Moreover our solution and the optimal geometry presented in [72] are very similar. The main difference comes from the length of the foundations that is fixed to 2 m in our application and is free (and smaller) in [72]. Compared to topology optimization where the interface between the dam and the water have to be interpolated, the use of a Level Set description is much easier as we have a clearly defined interface zone where the loads can be imposed.

## 6.7 Optimization of a impactor impeller

This industrial application aims at analyzing and providing new impeller designs for the Mag Impact, a high performance vertical shaft impactor designed and produced by the company Magotteaux represented in Figure 6.48.

We have to study the shape of the impeller blades that are fixed on the rotating table illustated in Figure 6.49 (a) and (b).

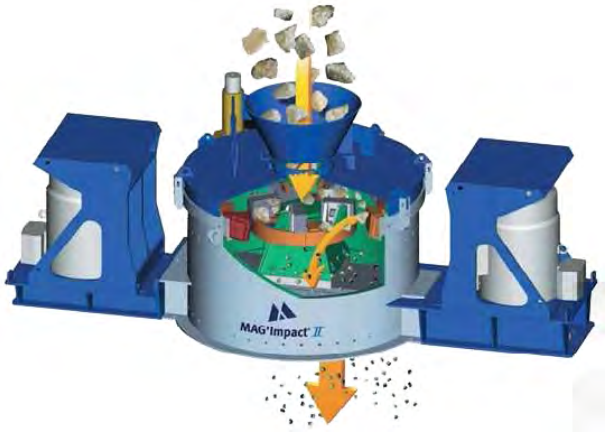



Figure 6.48: MAG Impact impactor

(a) MAG Impact table 3D view

(b) MAG Impact table (cross section)

Figure 6.49: MAG Impact table

### 6.7.1 Analysis of the initial 2D design

The impeller is studied on a 2D model derived from the original 3D CAD. The geometry considered is represented in Figure 6.50 and corresponds to a slice of the 3D impellers at the level of the table. The positions of the different points can be found in [55].

To simulate the presence of the ceramic padding, which ensures a high wear resistance, the geometry is divided in two domains. The upper rectangular domain in Fig. 6.50 is the padding zone. As it is the working zone of the impeller, it cannot be modified and is therefore kept fixed during the optimization. The lower domain (back face of the impeller) is submitted to the optimization process and is the design domain for the shape optimization. This domain will be referred as domain 2.

#### Material data and boundary conditions

Although the two domains are made of different materials, the mechanical properties assigned to the two domains are considered as equals. Indeed, according to the designers, the padding is supposed not to contribute to reinforce the impeller but is only present to improve the wear resistance. The mechanical properties considered are then: elastic isotropic material, Young

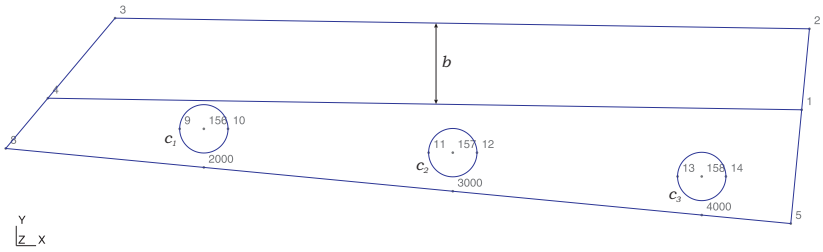



Figure 6.50: Geometry of the impeller in its original configuration

modulus of $E$=100 GPa, Poisson's ratio $\nu$=0.3 and mass density of $\rho$=7800 Kg/m$^3$.

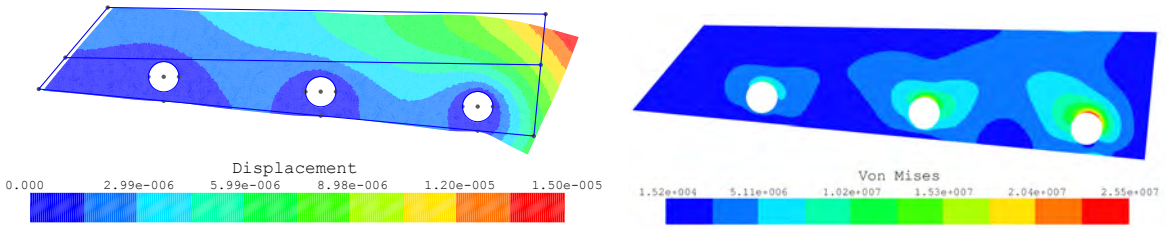
The 2D structure is supposed to be in plane stress state and the thickness is taken equal to the 3D thickness measured perpendicularly to the table. The considered load case is a centrifugal load $F$=$m\omega^2 r$ about a rotation center located at the table center. The rotation speed of the table is 1000 $rpm$. The structure is fixed at the 3 circles $C_1$, $C_2$ et $C_3$ (see Fig. 6.50), which corresponds to a perfect clamping of the impeller with infinitely rigid bolts.

Heavy assumptions have been made to build the model, which is justified by the preliminary character of this study. Ideally, the impeller should be studied in 3D and a pre-stress between the bolts and the impellers as well as the clearance which should affect the boundary conditions should be taken into account for further improvements. The impacts of the rocks with the impeller may also be introduced.

In order to compare the results obtained after a shape optimization, the structure is first analyzed in its initial configuration. The main difficulty of this model is that the fixation holes are not modeled with a conforming mesh but with three Level Set. So the boundary conditions have to be prescribed within the mesh. To this end, the geometry is meshed with linear triangle finite element and the Dirichlet boundary conditions are applied with the method presented in section 4.5.2.

Figure 6.51 (a) gives the displacements obtained with X-FEM. The maximum displacement is located at the upper right corner of the impeller and takes a value of $u_{max}$ = 14.95 $\mu$m. Due the eccentric position of the rotation center, the centrifugal load is not parallel to the axis $x$ of the impeller local referential (see Fig. 6.50). The loads has therefore a non zero $y$ component that tends to bend the external part towards the back of the impeller.

The Von Mises stresses for the initial geometry is given in Figure 6.51 (b). The most stressed zone is located around the fixation, at the external radius, where the centrifugal loads are the highest. The maximum values ($\sigma_{max}$ = 26 MPa) occur in the upper part of the fixation $C_3$. We can remark here the limits of the model considering a perfect clamping as the bolts should normally introduce some compression and traction efforts. Considering the St Venant Principle, we can admit that this modeling imperfection should only introduce local perturbations and not too many errors in the model.




(a) Geometry and displacements of the initial impeller geometry (m)

(b) Von Mises stress for the initial configuration (Pa)

Figure 6.51: Initial configuration

### 6.7.2 Optimization of the bolt positions

In a first step, the optimization tries to find the best positions of the bolt that attach the impeller to the table. The geometry of the impeller is fixed. and the design variables are the position $(x, y)$ of each bolt $C_1$, $C_2$ et $C_3$ of the model. The choice of these design variables calls for the application of variable embedded boundary conditions non conforming to the mesh. These circles are restricted to stay inside the design domain by introducing geometrical constraint. The objective function is to minimize the structural compliance. No mechanical or mass constraints are considered as the problem is self constrained in body load problem as noticed by Bruyneel and Duysinx in [40].

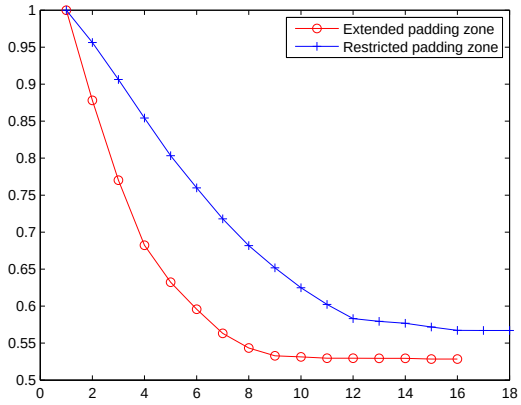



Figure 6.52: History of the objective function for the restricted padding zone and extended padding zone

In Figure 6.52, the blue cross curve represents the evolution of the objective function (relative to the initial value of the compliance). In this figure, blue cross curve refers to this first case named 'restricted padding zone' whereas the circle read curve is relative to the next example (see 6.7.2). Considering the positions of the fixation as design variables allows to increase the rigidity of the structure by 43 %. This gain is realized by moving the two outer fixations towards a new position which is more symmetric with respect to the thickness of the structure. Notice that the two bolts $C_2$ et $C_3$ move to the outer radius of the impeller whereas the first bolt $C_1$ (initially close to the padding) does not move significanlty. For all fixations, the $y$ positions is touching the geometrical constraints prescribed in order to take care of the presence of the padding. The

rigidity improvement is rather large and the maximal deformation of the impeller is reduced by 55 % ($u_{max}$=6,5 $\mu$m instead of 14.9 $\mu$m). The design improvement introduces a reduction of 37 % of the maximum Von Mises Stress compared to the intial geometry.

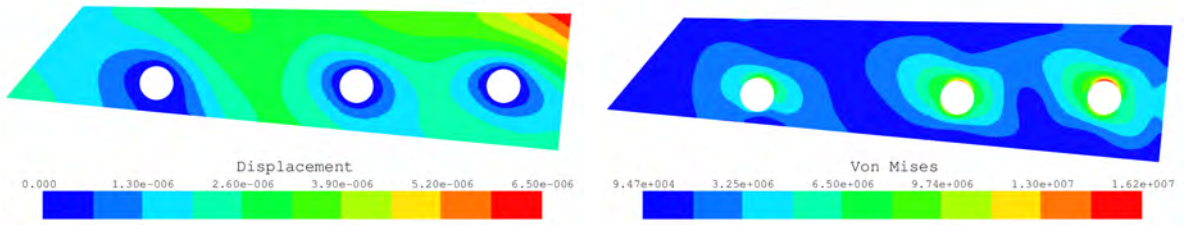



Figure 6.53: Displacement (m) and Von Mises stresses (Pa) after the optimization of the fixation positions considering the geometrical restriction for the padding

### Extended padding zone

Analyzing the worn impellers shown in Figure 6.54, we can remark that the padding zone is highly deteriorated in the peripheral zone at the proximity of the fixations $C_2$ and $C_3$ but it remains nearly undamaged in the upper left zone. This suggests that the geometrical design domain considered before could be extended in order to obtain a larger design domain especially around the fixation $C_1$.

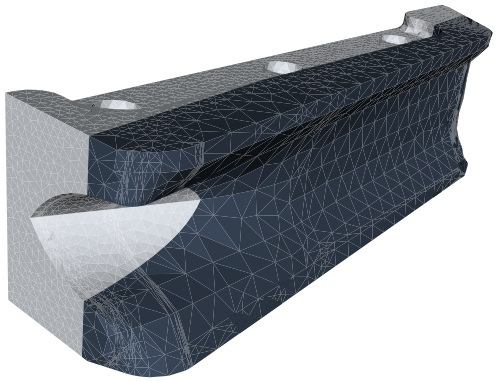


Figure 6.54: 3D model of a worn impeller

In the initial design, the padding zone presents a constant thickness that highly restricts the design domain where the fixation positions can evolve. In order to obtain more design freedom, especially for the fixation $C_1$, we redefine the padding zone by placing a parallel line (5-8) passing by the point 1 (see Fig. 6.55).

Hence, the outer radius thickness is kept equal compared to the initial domain, as it should not modify the life time of the initial impeller. The maximum thickness of the padding is equal to 58 mm at the outer radius and 52 mm at fixation $C_3$.

Replaying the optimization process with this larger design domain, the new optimal fixation positions are characterized by the displacement and Von Mises stresses in Fig. 6.56.

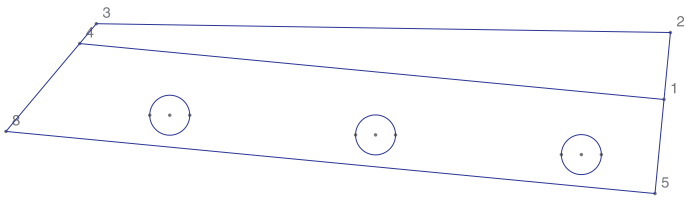



Figure 6.55: Geometry of the extended model with a variable padding restricted zone

By reshaping the design domain with a variable padding thickness, the solution of the optimization gains 6 % more in terms of stiffness (see Fig 6.52 red circle line). In this configuration, the optimal positions of the two fixations $C_1$ and $C_2$ are not touching anymore the gometrical constraint due to the padding restricted zone, the fixation $C_3$ still being limited by the padding zone.

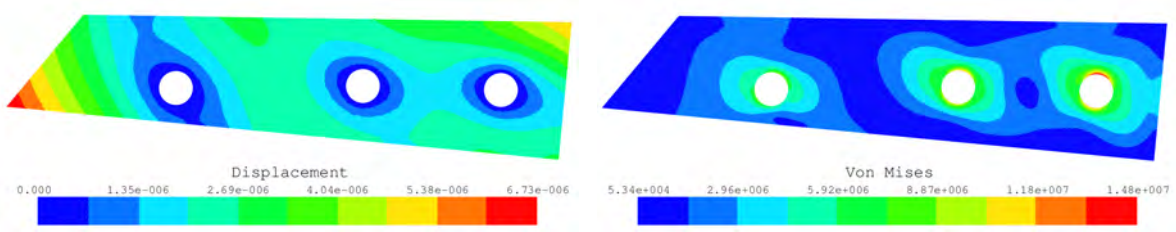



Figure 6.56: Displacement (m) and Von Mises stresses (Pa) of the extended optimized model

Finally, it is interesting to remark that this optimal geometry presents a maximum displacement $u_{max}$=6.7 $\mu$m, which is higher that with the initial padding domain (+ 5 %) but has a maximum Von Mises stress reduced by 10 %.

### 6.7.3 Optimization of the back face of the impeller

Finally, we consider that the back face of the impeller is the designable feature. To this end, we replace the straight line 8-5 of the initial geometry with a 9 control points Level Set spline in order to be able to modify the back face of the impeller (see Fig. 6.57). In this problem, the optimization variables are on the one hand the positions $x$, $y$ of the three fixations and, on the other hand, the 7 control points of the Spline curve.

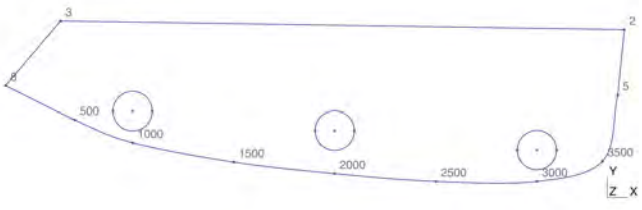



Figure 6.57: Initial modified geometry

The points {500,1000,1500,2000,2500,3000} are free to evolve along the $y$ axis whereas the point

3500 can move in both $x$ and $y$ direction. The points {500,1500,2500,3500} are constrained to remain outside of the padding zone and the points {1000,2000,3000} remain at the same abscissa as the center of the fixation circles. The $y$ coordinates of these ones should be greater than the coordinates of the opposite control points with a gap of 27 mm to take into account of the presence a the bolts and nuts. The two extreme points {8, 5} cannot be modified during the optimization procedure. All together, the problem has 12 design variables.

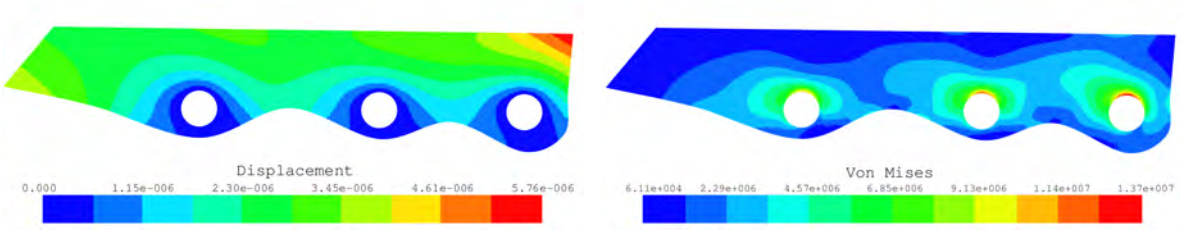



Figure 6.58: Displacement (m) and Von Mises stresses (Pa) in the optimal geometry

Compared to the previous optimal solutions, the flexibility of the Spline curve on the back of the impeller enables to obtain an important gain of performance simultaneously for the objective function, the maximum displacement and the maximum Von Mises stress. Hence, the structure presents a rigidity 33 % higher than the previous optimized model and 69 % higher than the inital design. The maximum displacement $u_{max}$ is equal to 5.7 $\mu$m to be compared with 14.9 $\mu$m for the intial geometry and the Maximum Von Mises stres is equal to 13.7 MPa whereas in the intial geometry it is equal to 26.1 MPa (see Fig. 6.58).

## 6.8 Design of a 3D Tuning Fork

This test case aims at finding the length of a tuning fork in order to obtain a prescribed frequency tuned at A-440 Hz. To do this, the design of the tuning fork (Fig. 6.59) has been parameterized to present a first natural frequency with a symmetric bending eigenmode (see Fig. 6.59).

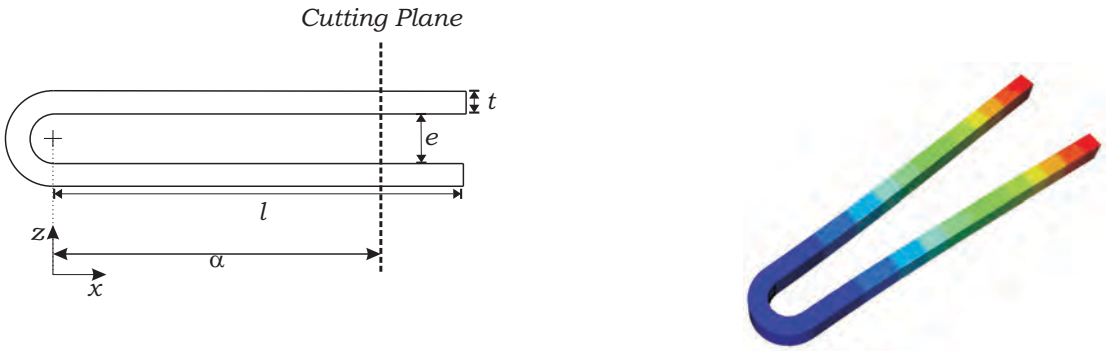
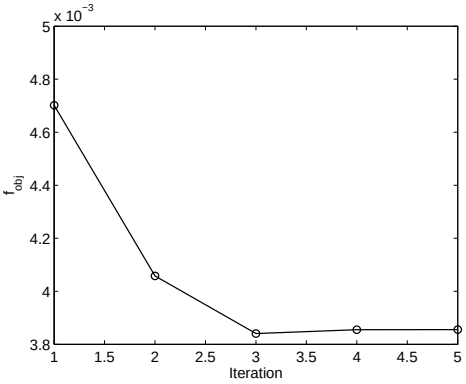



Figure 6.59: Geometry and first eigenmode of the tuning fork

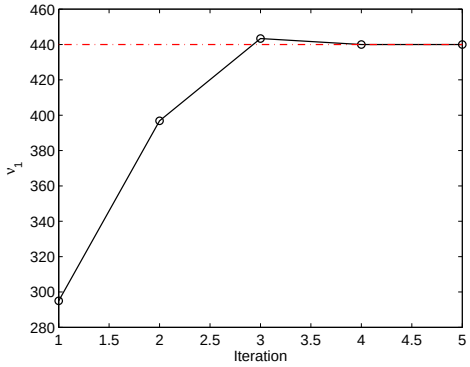
The initial geometry of the tuning fork is $l$=130 mm, $t$=3.9 mm and $e$=8.3 mm. The thickness is 5.5 mm and the material properties are: Young modulus $E$=2.11E11 N/m$^2$, Poisson's ratio $\nu$=0.3 and a density $\rho$=7800 kg/m$^3$. As the initial design presents a first natural frequency

of 295 Hz, by analogy with a guitar string, the simplest geometrical modification to obtain a higher first frequency consists in reducing the length $l$ of the fork. To this end, a cutting plane perpendicular to the $x$ axis is introduced (see Fig. 6.59) in order to be able to modify the length of the tuning fork by adjusting the variable $\alpha$ that define the position of the plane. The tuning fork is modeled with 62,000 linear tetrahedra and has free boundary conditions.

The optimization problem is first formulated as the minimization of the volume with an upper bound on the first eigenfrequency of 440 Hz. The initial value of $\alpha$ is set to 100 mm. Using CONLIN optimization, a solution presenting a first eigenfrequency at 440 Hz for a value of $\alpha$=80.27 mm is obtained in 5 iterations.



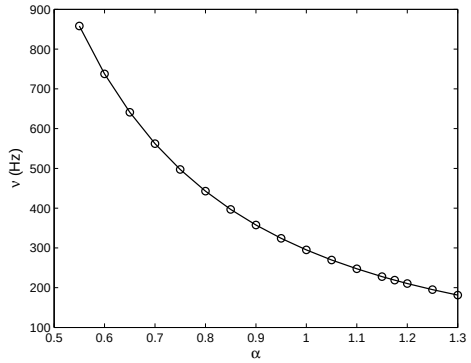

(a) Volume minimization history



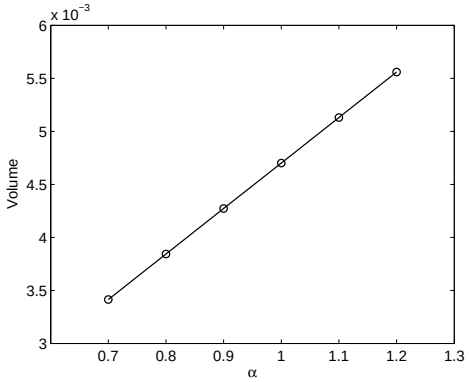

(b) History of the constraint on the first natural eigenfrequency

Figure 6.60: Objective function and constraint histories

The first optimization problem formulation is not general as it is based on the intuition that the frequency will increase when the variable $\alpha$ decreases leading to a reduction of the mass (volume) as it is be confirmed in Figure 6.61.




(a) Evolution of the first eigenfrequency with $\alpha$

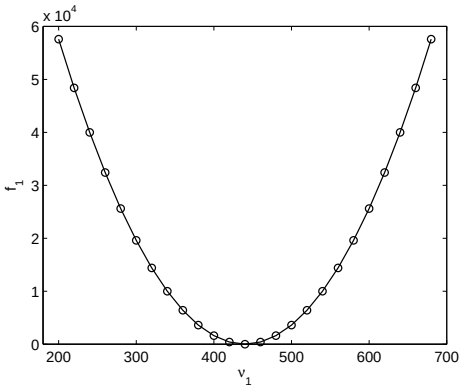



(b) Evolution of the volume with $\alpha$

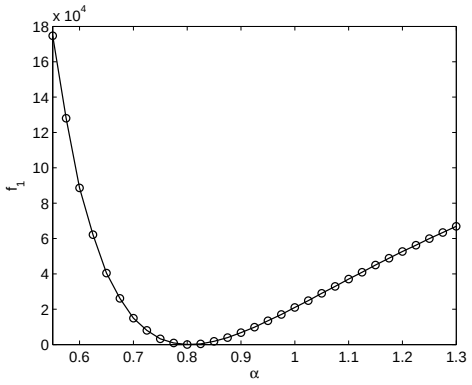
Figure 6.61: Parametric study of the first tuning fork

However, the parametric study presented in Figure 6.61 illustrates that the evolution of the first

eigenfrequency is similar to a function of the type $1/\alpha$ while the volume is linear in $\alpha$. Hence, these two functions are very well adapted to the CONLIN approximation which is confirmed by the small number of iterations necessary to come to the prescribed frequency.




(a) Objective function $f_1{=}(\nu_1 - \nu^*)^2$




(b) Parametric analysis of the objective function $f_1(\alpha)$

Figure 6.62: Objective function $f_1(\nu_1)$ and $f_1(\alpha)$

The same problem can be studied with a more general point of view by minimizing a function presenting a minimum located at the target $\nu^*$ value. To this end, we define a new objective function $f_1$ (represented in Figure 6.62) as:

$$f_1(\nu_1) = (\nu_1 - \nu^*)^2$$

where $\nu^*$ is the target value of the first eigenfrequency and $\nu_1$ the current value of the first eigenfrequency of the current design. The sensitivity of the objective function is readily obtained:

$$\frac{df_1}{d\alpha} = 2\,(\nu_1 - \nu^*)\,\frac{d\nu_1}{d\alpha}$$

posing $\lambda{=}\omega^2$, with $\omega$ the natural pulsation, we have:

$$\frac{df_1}{d\alpha} = -\frac{2\,(\nu_1 - \nu^*)}{8\pi^2\nu_1}\,\frac{d\lambda}{d\alpha}$$

Stating the optimization problem as the minimization of the function $f_1$, with $\nu^*{=}440$ Hz, without any constraint, we obtain the convergence curve sketched in Figure 6.63.
In Figure 6.63, it can be observed that the number of iterations to come to the target value (within the same tolerance than the previous formulation) is much higher inasmuch as 15 iterations are performed. Indeed, we can see that the function $f_1$, represented in Figure 6.62 (b), is non monotonous and CONLIN can hardly represent it, which leads to a high number of iterations.

To remedy to this problem, we can introduce a new objective $f_2$ better adapted to the CONLIN approximation by adopting a function based on the principle of the moving asymptotes of MMA:

$$f_2(\nu_1) = \frac{b}{\nu_1 - a}$$

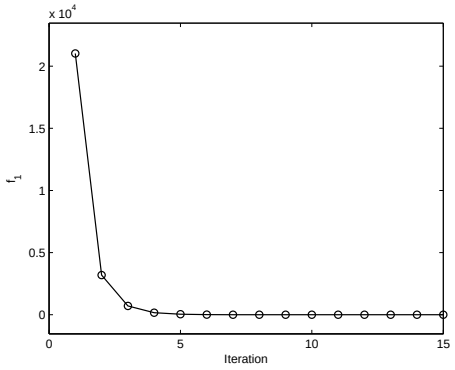
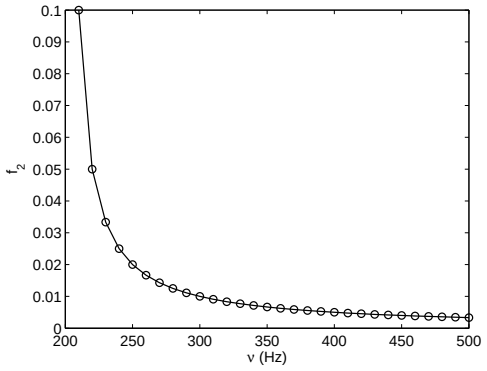
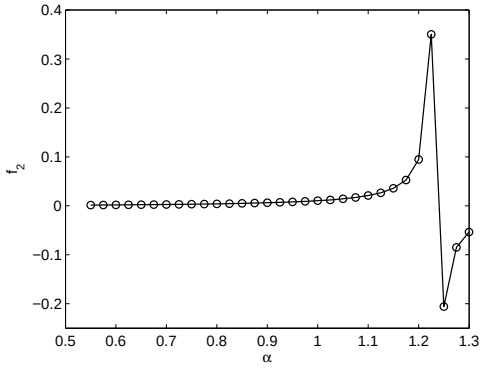



Figure 6.63: Iteration history for the minimization of $f_1{=}(\nu_1 - \nu^*)^2$

where the value of $a$ is chosen to introduce an asymptote in $\nu_1 = a$ and $b$ is set to +1 or -1 depending on the current value of $\nu_1$ ($b$ is positive when $\nu_1{<}\nu^*$). This function is represented in the Figures 6.64 with respect to $\nu_1$ and $\alpha$.




(a) Objective function $f_2(\nu_1){=}\frac{b}{\nu_1 - a}$ ($a$=200,$b$=1)




(b) Parametric analysis of the objective function $f_2(\alpha)$ ($a$=200,$b$=1)

Figure 6.64: Objective function $f_2(\nu_1)$ and $f_2(\alpha)$

The derivative is easily obtained in terms of the natural pulsation:

$$\frac{df_2}{d\alpha} = -\frac{1}{(\nu_1 - a)^2}\frac{d\nu_1}{d\alpha} = -\frac{1}{8\pi^2\nu_1\,(\nu_1 - a)^2}\frac{d\lambda}{d\alpha}$$

Compared to the function $f_1{=}(\nu_1 - \nu^*)^2$, the function $f_2{=}\frac{b}{\nu_1 - a}$ is monotonous and the minimum has to be bounded by the introduction of a constraint on the target value $\nu_1$. In this case, the optimum is achieved by the optimizer after only 4 iterations. The computational effort is much lower in this formulation compared to the formulation with one volume constraint as only the derivative of the frequency has to be computed.

Notice that the solution has been compared with FEM for the prescribed value of $l$=80.27 mm and

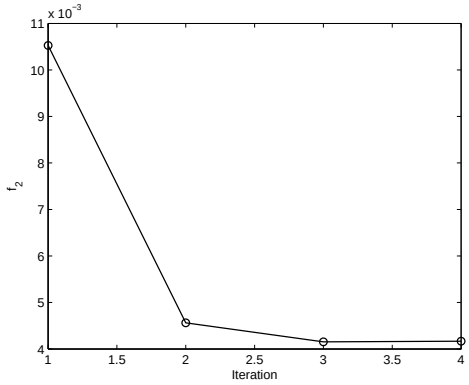



Figure 6.65: Evolution of the objective function $f_2{=}\frac{b}{\nu_1 - a}$, ($a$=200,$b$=1)

a very good agreement is obtained as the FEM model predicts a first eigenfrequency at 439.79Hz.

We observe that with this application our sensitivity analysis scheme can fail when the perturbation over the variable $\alpha$ is positive. In this case, there is a strong probability that new elements appear during the sensitivity perturbation step. To prevent the creation of new elements, we have to prescribe a negative step $\Delta\alpha$ for the computation of the semi-analytic scheme to avoid underestimation of the derivative.

| Sensitivity method | Sensitivity | Difference |
|---|---|---|
| Finite difference + | 8.09091E6 | 0.6% |
| Finite difference − | 8.18182E6 | 0.5% |
| Semi analytic + (6 elements are created) | 7.53568E6 | 7.5% |
| Semi analytic − (0 element is created) | 8.13194E6 | 0.1% |
| Semi analytic adapted +/− (0 element is created) | 8.15478E6 | 0.2% |
| Analytic | 8.14E6 | - |

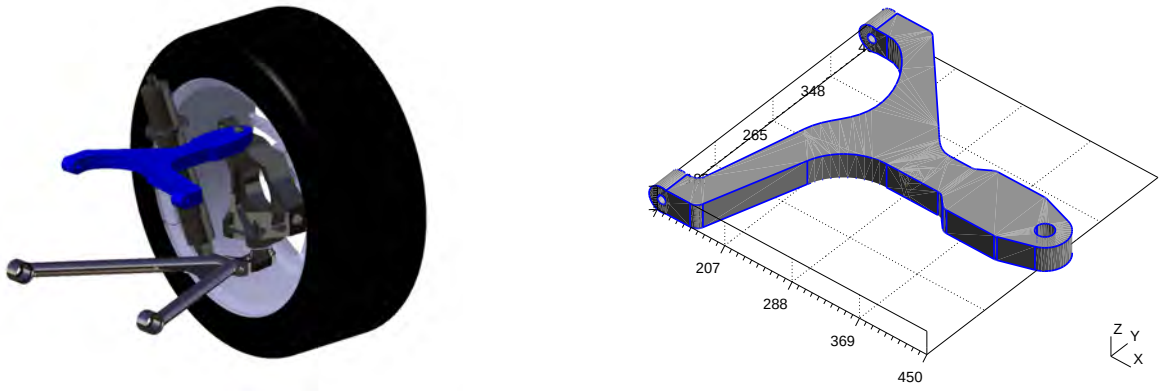
Table 6.4: Comparison of sensitivities with different approaches for $\alpha$=1.1

The Table 6.4 presents a comparison of the derivative of the first eigenfrequency for $\alpha = 1.1$ computed with different approaches, the '−' and '+' signs indicating whether the sensitivity is computed with a negative or a positive perturbation $\Delta\alpha$ step respectively. The mention +/− in the semi-analytic adapted method refers to the computation of elementary derivatives and the use of an adapted perturbation step $\Delta\alpha$. In this case, no new step has to be computed and the sensitivities are computed using a mixed $+\Delta\alpha$ and $-\Delta\alpha$ perturbation depending on the configuration of the element. Moreover, as this problem has only one variable, a single velocity field can be obtained analytically. The results obtained with an analytical method which serves as a reference for the comparison for the other approaches illustrates the accuracy of the implemented semi-analytic methods .

## 6.9 Shape optimization of a 3D suspension triangle

To illustrate the ability of our method to deal with 3D geometries as well as providing large shape modifications, we study the design of the arm of a double wishbone suspension car. The geometry of the suspension arm being dictated by the position of the kinematical joints between

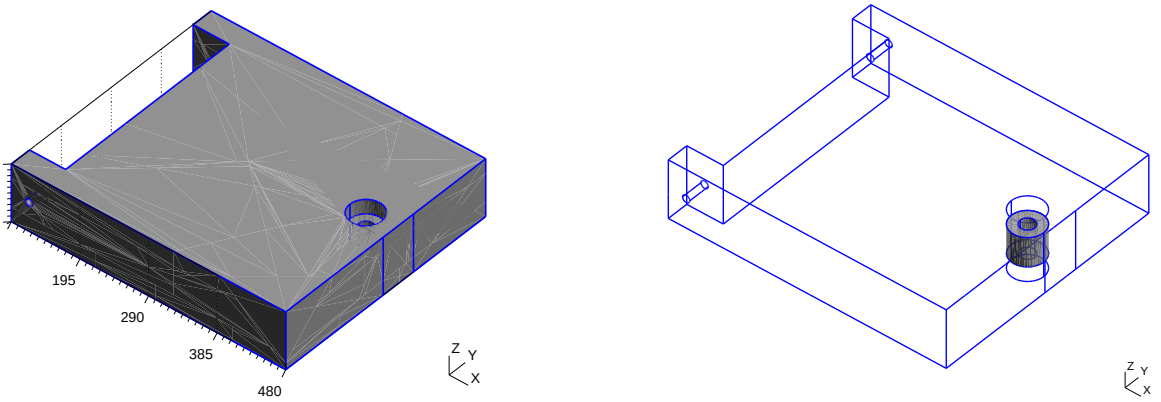
the arm and the car body, the upper arm have to be optimized without changing the boundary conditions. The complete suspension is represented in Figure 6.66 (a) and the suspension arm is shown in Figure 6.66 (b).




(a) Double wishbone suspension (b) Arm geometry (mm)

Figure 6.66: Initial suspension arm

To realize the optimization, an equivalent critical static load case is considered. This loading mimics the situation where a car of 1445 kg overcomes a succession of bumps in cornering. The bump travel is 15 cm and the speed of the car is 60 km/h. In the model, the applied load F is decomposed into 3 loads aligned along the x, y, z axis: the longitudinal load $F_x$=100 kN, the lateral load $F_y$=-28 kN and the vertical load $F_z$=62.5 kN. These loads are imposed on the right hole that connect the arm to the knuckle. Clamping fixations are applied on the two cylindrical holes which are the hinge connecting the suspension arm to the chassis. The initial structure design shown in Figure 6.66 has a compliance of 6.22E6 J.



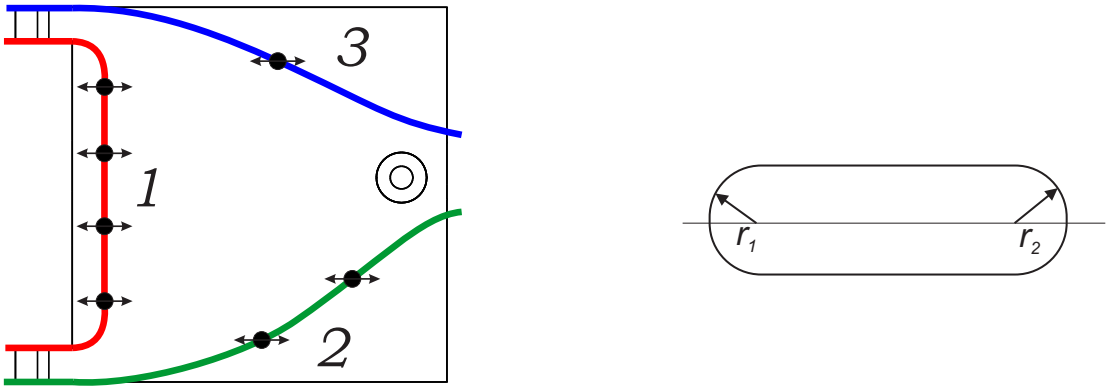

(a) Complete design domain (b) Non designable domain

Figure 6.67: Design domain

From the initial suspension arm, we can obtain all the geometrical data needed to create a suitable design domain. It is constructed to the bounding box of the actual suspension arm to which we allow a higher $z$ thickness compared to the initial geometry. This domain is represented in Figure 6.67 (a). It is composed of 2 volumes as illustrated in Figure 6.67 (b). The external

volume (transparent domain in the Figure 6.67 (b)) is composed of X-FEM elements and can be modified whereas the second volume (gray domain in Figure 6.67 (b)) is fixed. Additionally, we introduce in this domain the position of the fixations and loading surfaces on the joints. Both surfaces are discretized with a conforming mesh. This approach is chosen to facilitate the implementation of the model as we do not have to impose Neumann and Dirichlet inside the mesh. Moreover, given the constraint that the fixation and the loading domain are fixed, the choice of a conforming mesh is natural.

The geometrical description is built using four Level Set objects: three Level Sets representing a NURBS surface and one Level Set representing an oblong. One NURBS surface is placed between the two fixations to the car body (curve 1 in Figure 6.68 (a)) and the two others connect one fixation to the knuckle where the equivalent load is applied (curves 2 and 3 in Figure 6.68 (a)). The 3 boundary surfaces generated are perpendicular to the OXY plane and cut the entire design domain.




(a) Top view (b) Side view

Figure 6.68: Level Set parametrization

The black circles in Figure 6.68 (a) represent the control points of the different Level Set curves and the arrows indicate the direction of movement for the optimization. The last Level Set object represents an oblong (see Figure 6.68 (b)) extruded along the $y$ direction, which allows to modify the thickness along the $z$ direction.

The upper arm flexibility of the front suspension has an important effect on the vehicle handling during high-speed cornering. Therefore, objective function considers the minimization the structural compliance with a restriction on the total a mass that should be equal to the existing structure: 6 kg. The material considered has the following properties: Young modulus $E$=200 GPa, Poisson's ratio $\nu$=0.3. The problem involves 9 variables which are the 7 control points of the Level Set NURBS and the two radii of the oblong Level Set. Similarly to the 2D bracket support problem, we have to introduce several constraints on the design variables to prevent the degeneracy of the geometrical model.

The Figure 6.69 illustrates the evolution of the compliance and the volume of the structure. The red stars curve represents the volume, relative to its initial value. The blue crossed curve represents the compliance, relative to its value at the first iteration. The black dotted curve is the compliance of the structure relative to the compliance of the initial arm design. The history of the volume and compliance is typical of a compliance minimization problem. At the beginning,

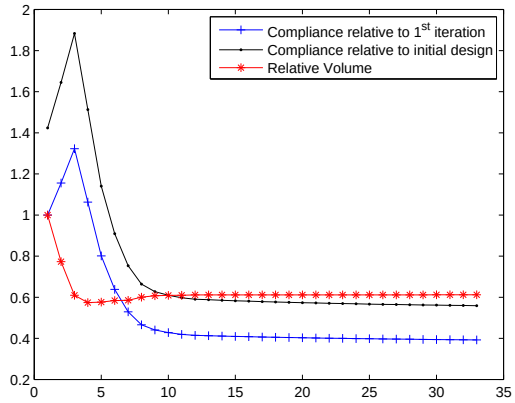
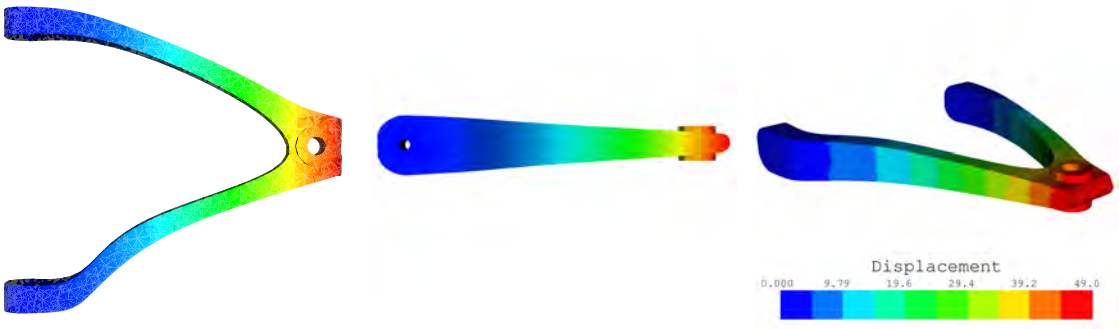



Figure 6.69: History of the objective function and volume constraint

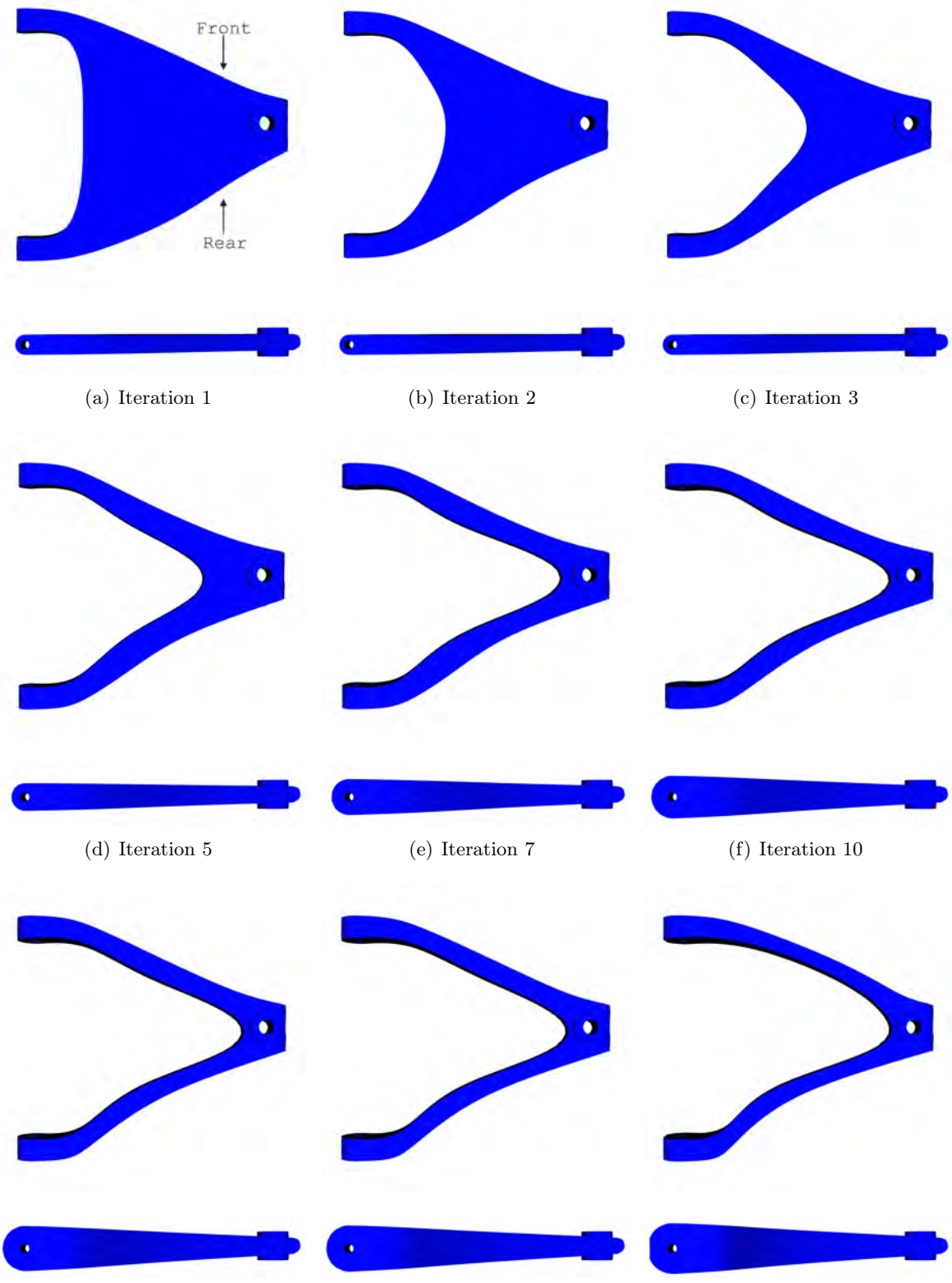
the volume is higher than the prescribed bound and the optimizer first reduces the volume yielding to an increase in the compliance. After two iterations, the volume satisfies the volume constraint and the compliance starts decreasing. Furthermore, we obtain after 5 iterations a new design which is stiffer than the initial geometry and the original arm. Then, approximately 5 more iterations are needed to obtain a significant gain in terms of rigidity. After this stage, the compliance does not change significantly and decreases at a slow rate. The optimal geometry (see Fig. 6.70) obtained, presents a 42 % stiffness increase with respect to the original design.




(a) Top view (b) Side view (c) 3D view

Figure 6.70: Optimal design

However, while the compliance does not evolve significantly after the $14^{th}$ iteration (less than 1% reduction between each iteration), some variables are still presenting a considerable variation between successive iterations. This is due to the fact that in the first iterations, the optimizer change the positions of the control points to meet the volume constraint in such a way that the movement of the Level Set 2 and 3 move to the left (see Fig. 6.71 (a)-(b)). In these geometries, both front and rear arms of the structure present a bended shape. Then, when the volume is lower, the optimizer progressively moves all the control points to the right and proceed to a fine tuning of the relative positions of the control points and of the oblong radii. Indeed, we can see in Figure. 6.71 (a)-(i) that the two arms become progessively thinner, the upper arm changing from a $S$ shape to a single curvature shape whereas the lower arm emphasizes its $S$ shape.




(a) Iteration 1 (b) Iteration 2 (c) Iteration 3

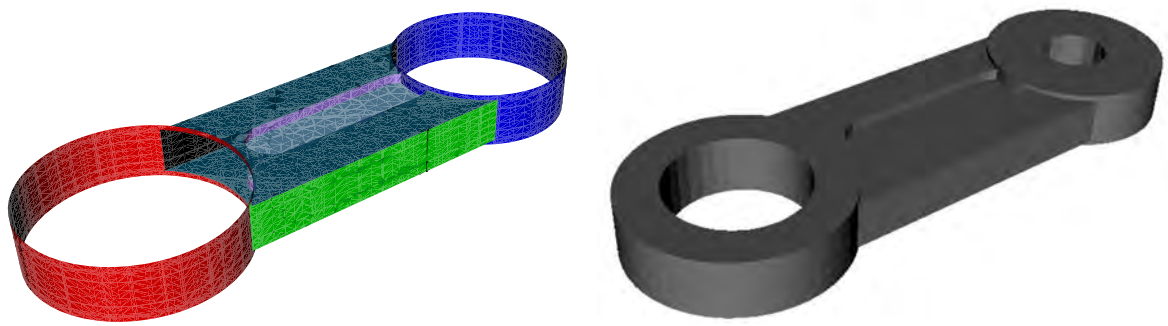
(d) Iteration 5 (e) Iteration 7 (f) Iteration 10

(g) Iteration 15 (h) Iteration 20 (i) Iteration 35

Figure 6.71: Optimal design history of shape optimization of an upper triangle suspension

## 6.10 3D Level Set CSG connecting rod

The final application considers the minimization of the weight of the connecting rod represented in Figure 6.72 (b). To obtain a rather complex geometrical representation that is close to an industrial design, the connecting rod is built with 12 Level Sets as illustrated in Figure 6.72 (a).



(a) View of the different iso-zero Level Set boundaries (b) View of the 3D initial design

Figure 6.72: Initial geometry of the connecting rod

The conrod is clamped at the big end connecting the left hole to the crankshaft. A compressive axial load F=60 kN is applied at the small end (right hole) which is connected to the piston pin. The stress constraint is limited to $\sigma_{eq} \leq$150 N/mm$^2$. The material has a Young modulus $E$=210E3 N/mm$^2$ and a Poisson's ratio $\nu$=0.3. The design variables are: the two external radii $r_1$, $r_5$, the depth $l_4$ and length $l_2$ of the central oblong pocket hole and the profile of the central arm that is modeled with two 3 control points NURBS surfaces (see Figure 6.73). The connecting rod is symmetric with respect to the horizontal axis and the oblong depth is symmetric with the rod thickness. The thickness is 18 mm on the central part and 22 mm at both ends. The initial geometrical parameters are: $l_3$=60 mm, $l_2$=60 mm, $l_4$=4 mm, $r_1$=40 mm, $r_2$=25 mm, $r_3$=12 mm, $r_4$=10 mm, $r_5$=20 mm and $r_6$=12.5 mm. The problem is studied with an analysis domain of size 250 × 100 × 22 mm meshed with 49728 linear tetrahedra. The meshing time with GMSH is 2.6 s for the 3D mesh and 16 s for the mesh optimization. By comparison, the time needed to slice the mesh is approximately 3 s.

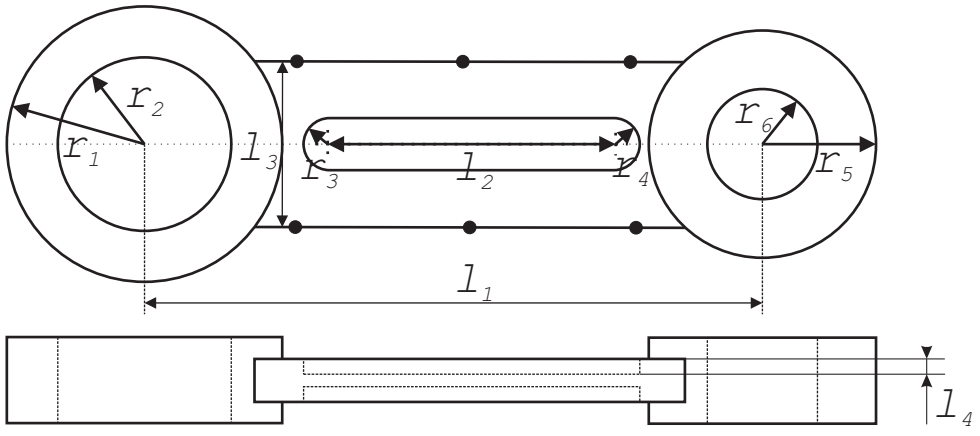
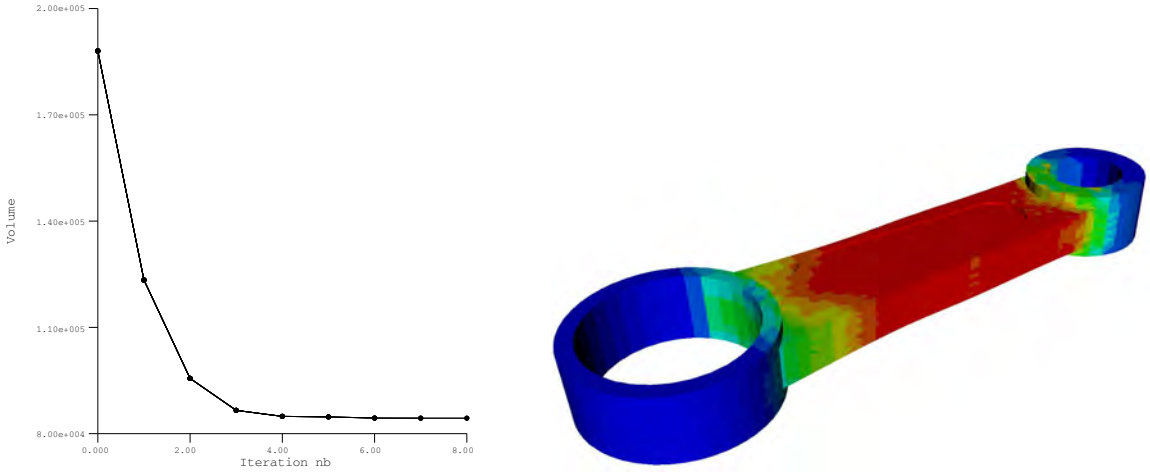


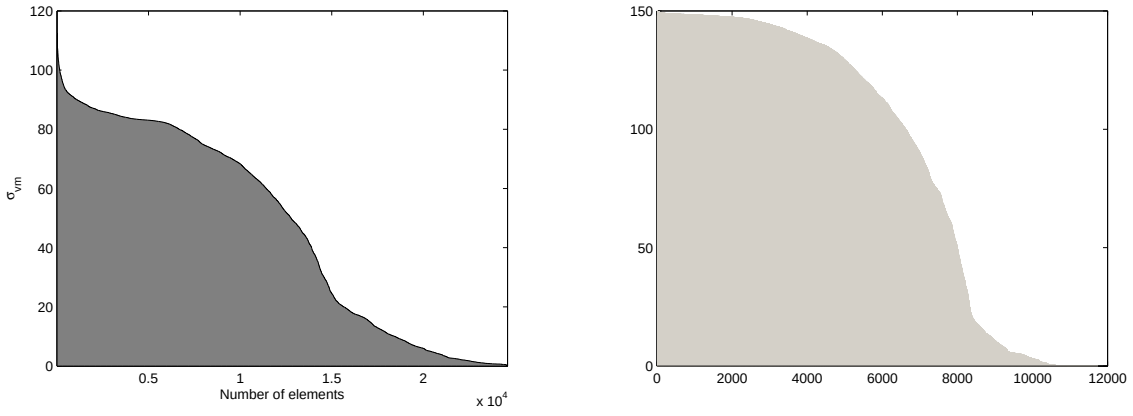

Figure 6.73: Conrod geometry and parametrization




(a) Objective function history (b) 3D view of final design with Von Mises distribution

Figure 6.74: History of the conrod volume and final geometry

The solution of this optimization problem is obtained after 8 iterations (see Fig. 6.74) and it yields a geometry where the central pocket hole has nearly disappeared, the two circles have a lower radius ($r_1$=30 mm and $r_5$=20 mm) and the central arm presents a higher thickness (18.4 mm compared to 18 mm) as illustrated in Figure 6.74 and 6.76. The volume reduction is equal to 222 % (from 188,025 mm$^3$ to 84,372 mm$^3$) and the maximum Von Mises stress increases from 112 N/mm$^2$ in the initial design to 150 N/mm$^2$ for the final configuration as prescribed by the constraints. The Figure 6.75 shows that in the final configuration the structure is widely stressed, indicating that the material is well used compared to the initial geometry where the Von Mises stress distribution is less uniform. Notice also that the total number of elements in the analysis is nearly divided by two from first iteration to the last iteration.




(a) Initial distribution (b) Final distribution

Figure 6.75: Von Mises stress distribution for initial and final design

The rod is studied considering only one load case that corresponds to a pure compressive case. In these conditions, the presence of the central pocket hole does not present any interest as it is

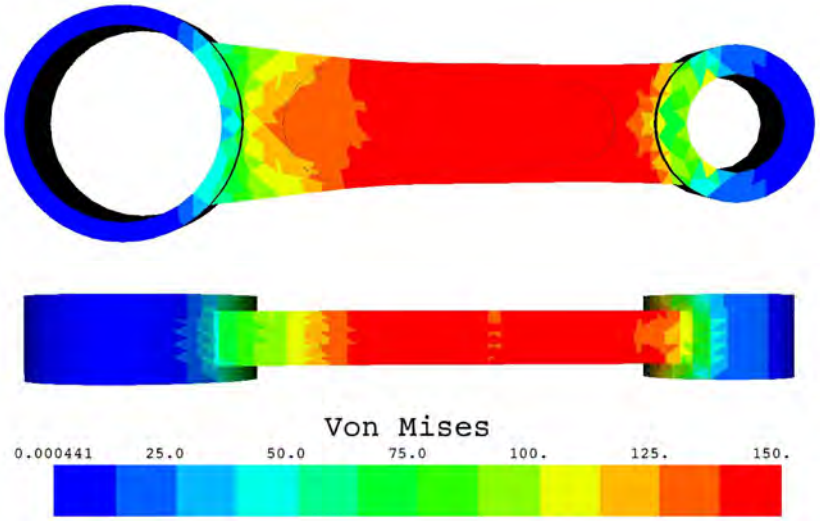



Figure 6.76: Optimal geometry with Von Mises stress distribution ($N/mm^2$)

the overall arm cross section which is important. However, if we observe a real connecting rod we can see that the central arm presents a $I$ or $H$ beam shape as illustrated in Figure 6.77.

Figure 6.77: Practial connecting rod design

The difference between our optimized shape and a real structure comes from the fact that the design process of a connecting rod has to take into account different load cases. The introduction of a multiple load cases optimization is therefore one important extension of the current development that could be done. Furthermore, in the case of the connecting rod, the compressive load is very high during the combustion stroke and one should not only consider the highest Von Mises stresses in the design process but also the buckling load factor. It is believed that introducing a constraint on the buckling load factor into the optimization procedure should provide a different geometry. Indeed, the depth of the central hole would be higher to achieve a significant improvement on the buckling load factor and provide $I$ or $H$ beam shapes.

Chapter 7

# Perspectives

The aim of this chapter is to present possible extensions of the current developments and further fields of applications where the X-FEM can be useful. The topics presented hereafter cover different preliminary studies realized with the aim of applying the X-FEM where we believe it may help the engineers to simplify existing approaches.

## 7.1 Optimization of periodic micro structures

When improving the performance of a structure, one has two choices: to modify the structure design (the shape or the topology) or to change the material properties. While the first approach has been studied since the beginning of the $60^{'}$s, the latter approach has received considerable attention with the advent of materials science, new composite materials, micro systems (MEMS) designs and new manufacturing processes such as the stereolithography. In this topic, the topology optimization has proven to be a nice tool to design new micro structures. In [130], Sigmund proposed the implementation of a topology optimization technique to obtain prescribed constitutive material properties using an inverse homogenization problem. Thereafter, researchers extended the range of applications to design new composite materials for instance with negative Poisson's ratio and also negative or zero thermal expansion coefficients that are very interesting for structure subject to temperature changes such as space structures.

In [151], in collaboration with Schoonjans, we have investigated the application of the X-FEM and Level Set description to realize preliminary investigations on design of periodic micro structures using the X-FEM code developed in this thesis. Several applications of finite and infinite sized structure have been realized with very promising results.

For instance, we have studied the shape of an optimal micro structure in an infinite plate under hydrostatic loading. Considering a thin elastic plate, the Vigdergauz's micro structure [157] is associated to the problem of finding the shape of spatially periodic inclusions that minimizes the potential energy.

A 2D infinite micro-structure is formed by replicating a basic cell that is considered for the analysis (see Fig. 7.1). This base cell is taken to be a square with a volume fraction denoted $c_2$ while the inclusion volume fraction is $c_1 = 1 - c_2$. In the present case, no second material is introduced and the inclusion is considered as a hole.

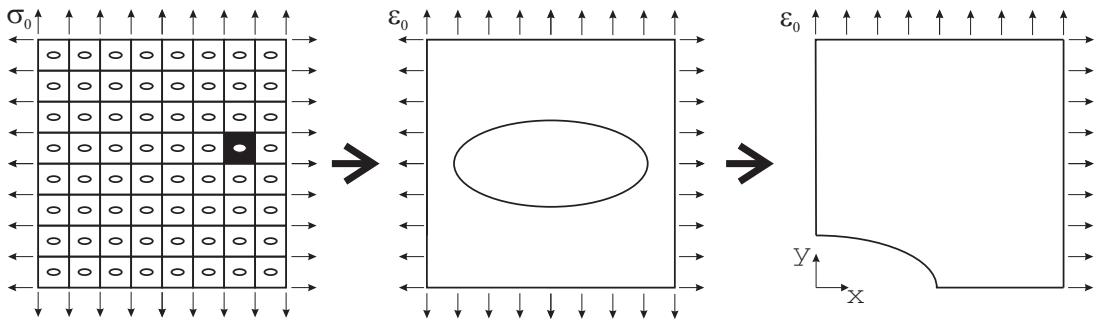



Figure 7.1: Modelization of a unit cell

The plate is remotely loaded by a uniform stress field $\sigma_0$: $\sigma_{xx}$=$P_0$, $\sigma_{yy}$=$Q_0$, $\sigma_{xy} = 0$. The behavior of the structure can be determined by isolating one cell and reproducing appropriate periodic boundary conditions. Considering a hydrostatic loading (Tr($\sigma_0$)=$\sigma_{xx} + \sigma_{yy}$=$\alpha$ and Dev($\sigma_0$)=$\sigma_{xx} - \sigma_{yy}$=0 with $\alpha$=1 N/mm$^2$), the optimal structure is symmetric and the displacement field is uniform over the side of one cell. This is achieved by imposing a displacement field $u$=$v$=1E-5 mm on the boundary. The cell has a side length of 5 mm and the inclusion boundary is represented with a super ellipse $\Gamma$=$\frac{x}{a}^{\xi} + \frac{y}{b}^{\eta}$. Since $u$=$v$, the solution is equal in $x$ and $y$ directions, the solution is such that $a$=$b$ and $\xi = \eta$ and the problem can be reduced by considering only 2 design variables. The range of design variables are: $a$=[0.5;4.5] and $\eta$=[1;20]. The initial conditions are $\eta = 4$ and $a$ is taken such that the volume ratio $c_1$ is achieved.

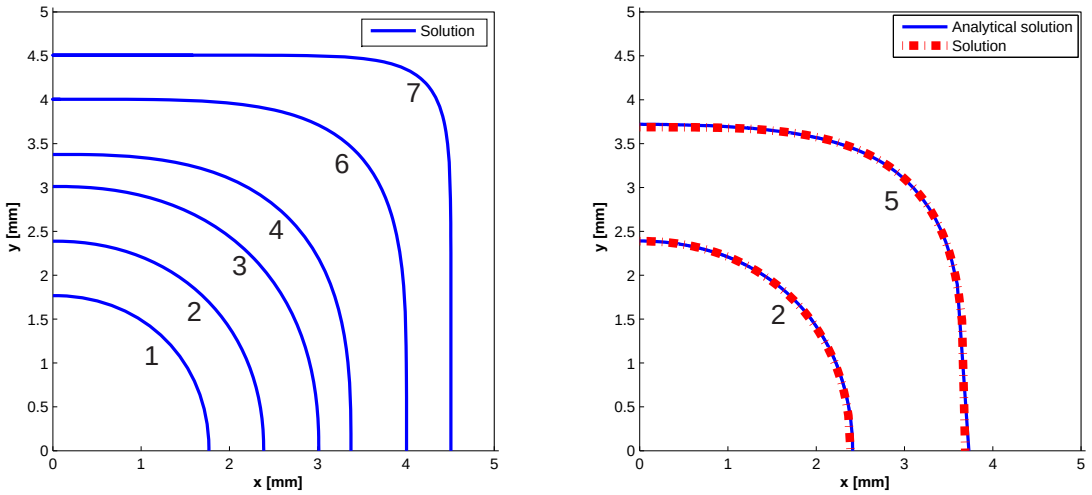



Figure 7.2: A quarter of a square cell: optimal shapes and comparison between analytical and X-FEM optimization for hole volume fraction of $c_1$ = 0.1 (1), 0.2 (2), 0.3 (3), 0.4 (4), 0.5 (5), 0.6 (6), 0.8 (7)

To obtain the analytical solution of Vigdergauz, we formulate the optimization as the minimization of the Von Mises stresses over a periodic cell with a prescribed amount of volume $c_1$. Repeating this problem for $c_1 = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.8$, we obtain the optimal holes represented in Figure 7.2. Until $c_1 = 0.4$ the shape is a smooth circular curve. For higher values of $c_1$, the exponent of the super ellipse expression grows rapidly to present a shape with nearly rectilinear sides and rounded corners. These holes appear to be very close to the analytical results of Vigdergauz [157] as indicated in Figure 7.2 right.

## 7.2 Stochastic Analysis

In the recent years, non-deterministic finite element methods have received an increasing attention. These methods intend to evaluate the effect of numerous uncertainties on the response of the structural system. These uncertainties can have several origins and arise either due to the lack of knowledge of some parameters and loading conditions or due to inherent variabilities of physical quantities. In non deterministic numerical methods, uncertainty sources are of various types:

- mathematical modeling assumptions;
- discretization error of discrete methods such as FEM;
- model parameters variation: material properties or loads uncertainties, manufacturing tolerances variability, ...

With the development of MEMS devices, the geometrical uncertainties related to the manufacturing tolerances and material properties are very important as it may severely affect the performance of such a device. For instance, the shape of the fixation of a clamped-free electromechanical beam will highly impact the MEMS structural response. Due to the large magnitude of uncertainties relative to the physical scale and highly nonlinear behavior of small-scale structure, it becomes very important to take into account geometric uncertainties in the design methodologies. While material properties uncertainties can be treated with limited computational effort within FEM, the geometrical uncertainties suffers from the same difficulties than classical shape optimization, which are the management problems and the associated computational cost. Regarding this aspect, one can conceive that the advantage of X-FEM in such approach can be helpful. Hence, when using a Monte-Carlo approach to compute the influence of a geometrical parameter, X-FEM can provide a fast re-analysis with less burden than if a FEM approach is used.

Additionally to X-FEM, the Level Set approach appears very interesting for treating geometrical uncertainties because it provides a direct method to generate a set of random geometries from a given nominal geometry. Reminding that one can freely modify a boundary represented with a Level Set description, the perturbed boundaries can be realized by using random velocity field $V_n$ as illustrated in Figure 7.3.

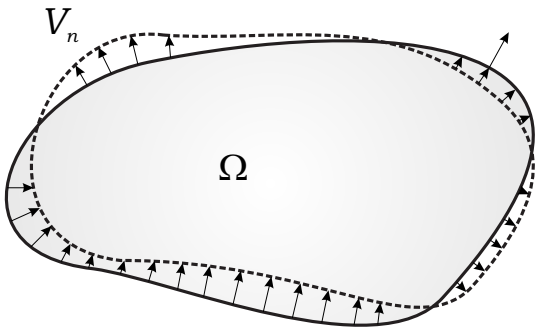



Figure 7.3: Nominal geometry and perturbation

These capabilities have already been used by Jang *et al.* in [87] where the geometry of micro compliant mechanisms are represented with uncertain etching parameters around a nominal model.

Because the boundaries of a structure are given as contour lines with a fixed nominal Level Set, the under- and over-etched structure can be constructed by shifting the level-set function uniformly upward and downward respectively. Because the uncertainties during fabrication process of micro systems are relatively large compared to macroscopic structures, their performance is highly sensitive to geometric parameters. Thus it is a primary importance to obtain reliable designs. For example, the hinges of micro compliant mechanisms play a key role in the kinematics of such device that can be severely affected by the etching uncertainties.

### 7.2.1 Stochastic Finite Element Methods

Stochastic finite element methods can be classified in two main categories depending on the type of results they yield. On one hand, reliability methods are interested in determining the failure probability and focus only on the probability density function of the response. On the other hand, other methods aim at computing the probabilistic characterization of the response.

In this category, particular methods that determine only the first two statistical moments of the response are available. In the preliminary study that we have carried out with Dr. Séverine Lepage, we have limited our investigations to two methods: the Perturbation Stochastic Finite Element Method (PSFEM) and to the Monte-Carlo method (MC). These two approaches are briefly presented hereafter. For more details about these methods. Interested reader should refer to [92, 99].

#### Monte-Carlo Simulations

The Monte-Carlo approach consists in generating a sampling for each uncertain input parameters, according to their probability distributions and correlations, and to proceed to a deterministic finite element analysis for each input sample. Then, given the corresponding response sampling, one can compute the mean and the standard deviation of the response. Monte-Carlo methods present the advantage that if a large number of simulations is carried out, they statically converge to the correct distribution but on the other hand it is highly computationally demanding due to repeated analyses.

#### Perturbation Stochastic Finite Element Method (PSFEM)

The Perturbation Stochastic Finite Element Method consists in realizing a deterministic analysis of the nominal structure complemented by an additional sensitivity analysis with respect to the random variables. This approach allows the development of a Taylor series expansion of the response field from which the mean and variance of these quantities can be derived according to the mean, variance and correlation of the random parameters. The main advantage of the PSFEM is its simplicity and its applicability to a wide range of problems with a limited additional computational cost. In this work, PSFEM approach appears to be the most interesting one as we can take advantage of the sensitivity analysis methods developed in this thesis.

### 7.2.2 Application: clamped - clamped silicon beam

In this application, we study the influence of the variation of the first eigenvalue of a clamped-clamped beam under an uncertainty of the beam height $h$. The geometry and the material properties are characterized by the following average parameters: Length=L=80 $\mu$m, Height $h$=4.5 $\mu$m, Young Modulus $E$=158E3 N/mm$^2$, mass density $\rho$=2.3E-6 kg/mm$^3$, Poisson's ratio

$\nu$=0.2.


Figure 7.4: Half of clamped-clamped beam geometry

The beam being in clamped-clamped conditions, only one half can be modeled by applying the suitable boundary conditions. As mentioned, the variable parameter is the height $h$, which distribution is: mean value=4.5 $\mu$m. The analysis domain is defined as a rectangle of dimension 40 $\mu$m $\times$6 $\mu$m modeled by 20 $\times$ 6 bilinear quadrangular finite elements.

***Monte-Carlo method***

The Monte Carlo solution is taken as a reference solution and it is computed with a population size $n$=500. The relative mean value for the first eigenvalue obtained is equal to 1.5012E15 $(\text{rad/s})^2$ and the standard deviation is 1.31163E14 $\text{rad}/s^2$.

***PSFEM***

In order to obtain the stochastic eigenvalue equations, the dependence on the uncertainty '$b$' for each parameter is introduced into the weak form of the governing equations. The stochastic finite element eigenproblem to be solved is then obtained by introducing the uncertainty dependence in the discrete eigenvalue equation:

$$K(b)q(b) = \lambda(b)M(b)q(b) \tag{7.1}$$

In practice, the uncertainty is represented by a finite set of random variables:

$$\mathbf{b} = \{b_1, b_2, b_3, ..., b_n\}$$

The perturbation method supposes that the random design variables $b_i$ can be decomposed as a sum of a deterministic value $\bar{b_i}$ and a zero mean random perturbation variable $\Delta b_i$:

$$b_i = \bar{b_i} + \Delta b_i$$

In the PSFEM approach, the random quantities in equation (7.1) are expanded around their expectation using a Taylor series expansion. For instance, the mass density $\rho$ can be approximated as:

$$\rho(b_1, ..., b_n) \approx \rho(\bar{b_1}, ..., \bar{b_n}) + \sum_{i=1}^{n} \rho_{,i}\Delta b_i + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \rho_{,ij}\Delta b_i \Delta b_j$$

where the subscripts $,i$ and $,ij$ respectively denote the first and second order partial derivative with respect to $b_i$ and $b_j$. Substituting the truncated Taylor series expansion into the finite element formulation (7.1) leads to the following eigenproblems to be solved:

- zero order term:

$$\bar{K}\bar{q} = \bar{\lambda}\bar{M}\bar{q}$$

- first order term ($i$=1,...,n):

$$(\bar{K} - \bar{\lambda}\bar{M})q_{,i} = (\bar{\lambda}M_{,i} + \lambda_{,i}\bar{M} - K_{,i})\bar{q},$$

- second order term ($i, j$=1,...,n):

$$(\bar{K} - \bar{\lambda}\bar{M})q_{,ij} = (\bar{\lambda}M_{,ij} + \lambda_{,ij}\bar{M} + 2\lambda_{,i}M_{,j} - K_{,ij})\bar{q} + 2(\bar{\lambda}M_{,i} + \lambda_{,i}\bar{M} - K_{,i})q_{,j},$$

with the superscript $\bar{.}$ denotes that the quantity is computed at the nominal value of $\bar{b}$.

Reminding that the probability that the value of the variable $X$ lies within an interval $[a, b]$ is given by:

$$P(a \leq X \leq b) = \int_a^b f_X(x)dx$$

where $f_X(x)$ is the probability density function of $X$, and the expectation of the function $g(X)$ is defined as:

$$E\left[g(X)\right] = \int_{-\infty}^{\infty} g(X)f_X(x)dx$$

The eigenvalue expectation is then given by:

$$\begin{aligned} E\left[\lambda(\bar{b})\right] &\approx \bar{\lambda} + \sum_{i=1}^{n} \lambda_{,i}\Delta b_i + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \lambda_{,ij} E\left[\Delta b_i, \Delta b_j\right] \\ &= \bar{\lambda} + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \lambda_{,ij} Cov(b_i, b_j) \end{aligned}$$

since the $\Delta b_i$ are zero-mean random variables of known covariance. The eigenvalue variance is:

$$Var(\lambda(\bar{b})) \approx \sum_{i=1}^{n}\sum_{j=1}^{n} \lambda_{,i}\lambda_{,j} Cov(b_i, b_j)$$

where the covariance $Cov$ of two variables $b_i$,$b_j$ is defined as the expectation of the product of the deviations from their respective means, $m_i$ and $m_j$:

$$Cov(b_i, b_j) = E\left[(X_i - m_i)(X_j - m_j)\right] = E\left[X_i X_j\right] - m_i m_j$$

The approach is summarized in the Flow-Chart 7.5. For the silicon beam problem, we obtain a first mean eigenvalue of 1.4983E15 $(\text{rad/s})^2$ with a first order approach and 1.5012E15 $(\text{rad/s})^2$ with a second order approach to be compared with a value of 1.5012E15 $(\text{rad/s})^2$ for the Monte-Carlo method.

| Method | Mean $[(\text{rad/s})^2]$ | $\epsilon_m$ [%] | Std. dev. $[(\text{rad/s})^2]$ | $\epsilon_{std}$ (%) | Mean($\epsilon$)[%] |
|---|---|---|---|---|---|
| Monte-Carlo | 1.5012E15 | - | 1.3163E14 | - | - |
| PSFEM ($1^{st}$ order) | 1.4983E15 | -0.1894 | 1.3173E14 | 0.0733 | -0.1965% |
| PSFEM ($2^{nd}$ order) | 1.5012E15 | 0.0015 | 1.3173E14 | 0.0733 | 0.0035% |

Table 7.1: Mean value and standard deviation of $\lambda$ with MC and PSFEM

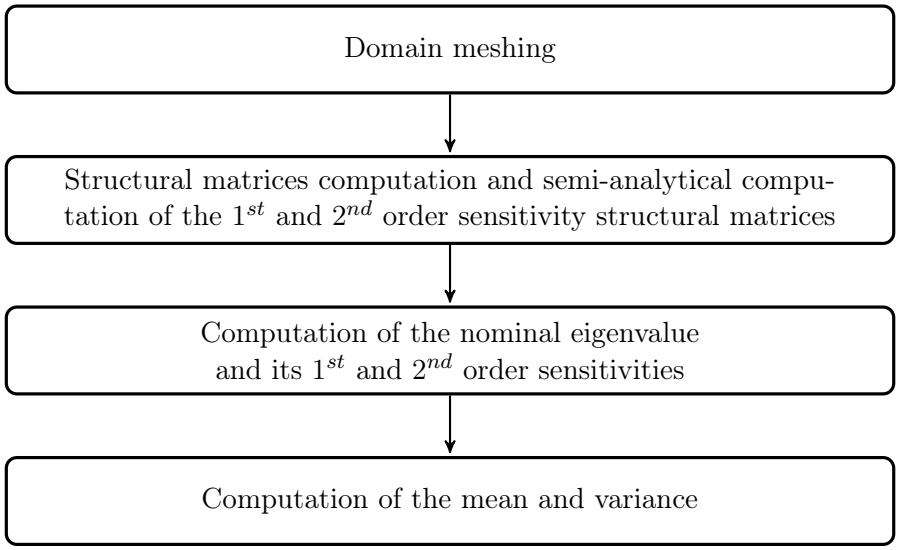
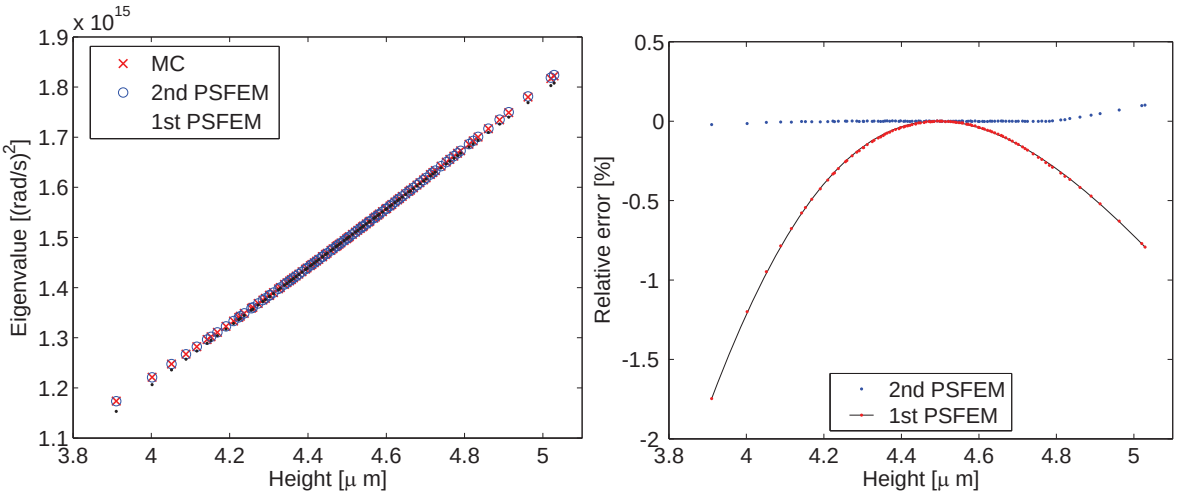



Figure 7.5: PSFEM Flow chart




(a) Eigenvalue variation with MC, PSFEM $1^{st}$ and $2^{nb}$ order

(b) Relative error of PSFEM

Figure 7.6: Comparison of Monte-Carlo and PSFEM results

We can observe in Figure 7.6 (a) that the PSFEM approach gives very good results compared to Monte-Carlo method. Indeed, using a $1^{st}$ order PSFEM, we can see in Figure 7.6 (b) that the relative error is very limited for small variation but increases quite quickly far from the nominal height value. Resorting to a second order PSFEM, we obtain an excellent relative error that is lower than 1% for a large range of beam height. Finally, the Table 7.1 summarizes the comparison of the MC and PSFEM method for this application.

## 7.3 Computation of pull-in with X-FEM

Accurately predicting the electrostatic forces appearing on (dielectric or conducting) moving structures is primordial in the modeling and simulation of advanced micro sensors such as RF-switches. In [143], a consistent variational approach was introduced to derive a monolithically coupled electromechanical Finite Element approach allowing accurate computation of pull-in [152] limits and dynamic behaviors. However, in the standard Finite Element techniques applied to coupled electro-mechanical models, one often has to face some serious limitations when the structure undergoes large displacements: the electric mesh around the structure is strongly deformed in order to follow the displacement of the structure. In this process, the electrostatic mesh may be severely distorted, as illustrated in Figure 7.7, and might, in the limit case of contact, rapidly degenerate.

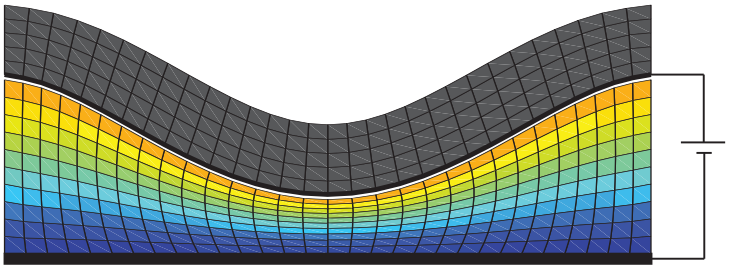


Figure 7.7: Deformation of the FEM mesh in a clamped-clamped micro beam under electrostatic field

To overcome this limitation, one can apply re-meshing or use the Boundary Element Method (BEM) in the electrostatic domain. However, both methods suffer from at least one drawback: it is not possible to compute the solution up to contact between the micro beam and the lower electrode. Here we investigate the possibility to represent the shape modification of the electrostatic domain using a fixed mesh method and to apply the X-FEM to account for boundaries and interfaces traveling across this fixed mesh.

In the electromechanical problem under consideration, the moving interfaces appear at the boundary between the electrical and mechanical domains while the latter undergoes large displacements through the fixed global electrostatic domain. The coupling occurring between electrical and mechanical fields comes from electrostatic forces acting on the structure. Those forces being themselves dependent on the deformation of the structure.

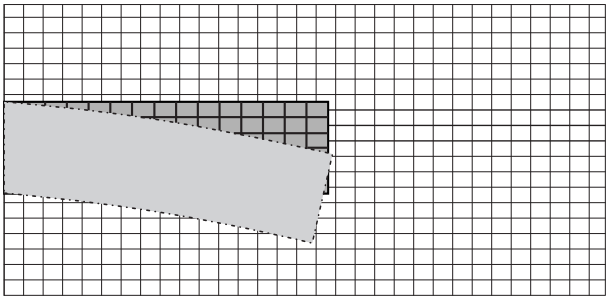


Figure 7.8: First approach, Lagrangian structural mesh and Eulerian air mesh

Two different approaches can be applied to take advantage of X-FEM for this kind of problems:

1. One can choose to model the beam using a FEM mesh that overlaps a second fixed X-FEM mesh modeling the air surrounding the beam as shown in Figure 7.8. In this case, we have two overlapping meshes and the electrostatic loads have to be projected from one domain to the other;

2. Alternatively, one can model the structure and the air using one Lagrangian mesh as in Figure 7.9 and let the mesh overlap the electrode, which is fixed and modeled with embedded boundary conditions.

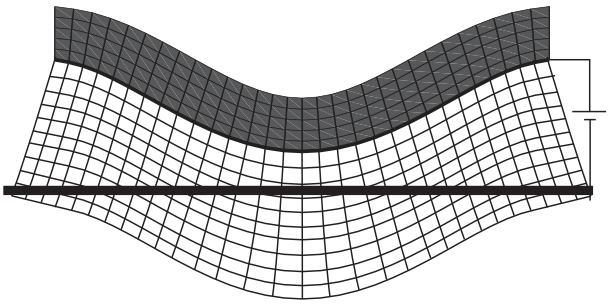


Figure 7.9: Second approach, Lagrangian mesh and fixed electrode modeled with embedded boundary conditions

The first approach is very general because there is no limitation on the shape and the movement of the structure or even on the number of structures overlapping the air domain. However, the challenge is obviously to properly handle the fact that some elements of the electrostatic domain are crossed by the interface. Moreover, this interface should be considered as a boundary condition for the electrostatic domain if the moving structure is a perfect conductor. This method has been tested in our implementations but it appears that the management and the computation of the electrostatic loads is very difficult. Indeed we have to deal with a high number of overlapping patterns in the projection of the loads which is quite difficult in 2D and even more arduous in 3D.

The second approach is much easier to implement because the computation of the loads does not call for a special treatment. The difficulty lies in the imposition of the Dirichlet boundary conditions inside the mesh to model the electrode. However, this approach is not general inasmuch the electrode should be fixed. Indeed, if we have two moving objects that get closer to each other, this method can not be used unless the analysis is decomposed into several sub domains. However, for simple geometries, this approach can be directly extended to 3D.

Currently, the second approach has been successfully investigated with two preliminary analyses. The first one consists in the computation of the equilibrium position of a clamped-clamped quartz micro beam subject to a voltage difference of 200 V applied between a lower electrode and the beam as illustrated in Figure 7.10. The geometrical parameters are: beam length $L$=30 $\mu$m, beam height $h$=1 $\mu$m, air gap (air domain between the lower and upper electrodes)=1 $\mu$m. The material properties are: air permittivity $\epsilon_{air}$=8.85E-012 F/m, beam Young modulus $E$=86.790E9 N/m$^2$, beam Poisson's ratio $\nu$=0.19 and beam permittivity $\epsilon_{beam}$=8.85E-9 F/m.

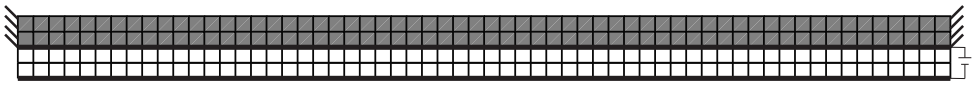


Figure 7.10: FEM mesh model

Using a Newton method with FEM to compute the equilibrium position, we obtain a displacement max($u_y$)=-5.35275E-2 $\mu$m for the central lower node of the beam (see Fig. 7.11).

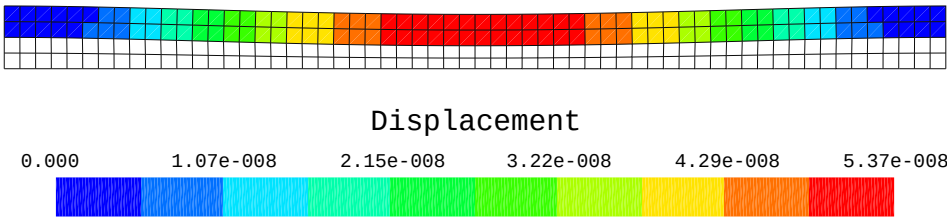



Figure 7.11: FEM equilibrium position. Mesh deformation scaled 5×.

For the X-FEM model, the geometry of the beam is the same but the air domain is extended at the bottom (see Fig. 7.12). The lower electrode is modeled by imposing a 0 V condition inside the mesh using a Lagrange multiplier method coupled to a Laplace stabilization as proposed by Bricteux and Remacle [39].

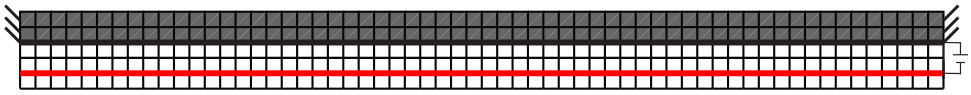


Figure 7.12: X-FEM mesh with non-conforming lower electrode

Whereas the electrode is modeled with strong boundary conditions and is clamped with FEM, the X-FEM approach enables the air domain to deform following the movement of the beam as illustrated in Figure 7.13.

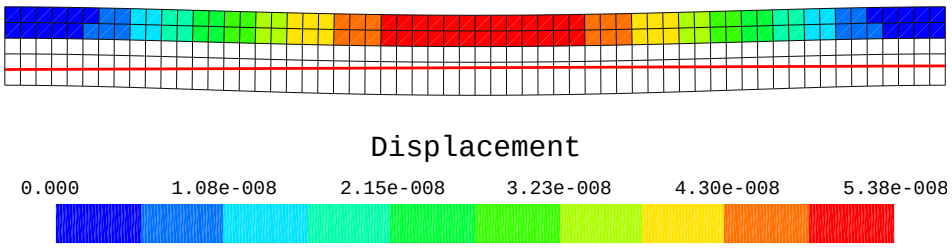



Figure 7.13: X-FEM equilibrium position with deformed mesh with lower electrode in red

Using X-FEM for the air domain, the equilibrium displacement of max($u_y$)=-5.35599E-2 $\mu$m compares very well with the FEM results (0.06 % of relative error with respect to FEM).

The second analysis considers the same geometry but we compute the pull-in voltage that is the limit voltage for which an equilibrium position can be found.

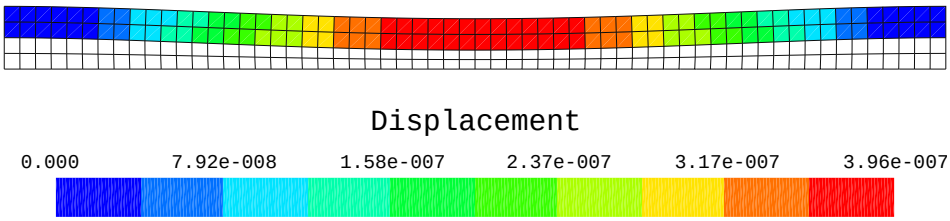



Figure 7.14: FEM pull-in equilibrium position. Mesh deformation scaled 1×.

The analysis is firstly realized using FEM [143]and the pull-in voltage computed is $V_{pi}$=399.32 V. The vertical displacement of the central node is equal to max($u_y$)=-0.39474 $\mu$m. In Figure 7.14 representing the deformed beam at $V_{pi}$, we can see that the mesh becomes thin at the center of the beam.

Using X-FEM, we obtain $V_{pi}$=398.88 V and the maximum vertical displacement is equal to max($u_y$)=-0.39628 $\mu$m. The relative error compared to FEM is 0.1 % for the pull-in voltage and for the maximum vertical displacement the relative difference is 0.4 %. The beam is represented at pull-in voltage in Figure 7.15. In this figure, we can see that the air mesh follows the deformation of the beam and the elements keep a good aspect ratio.

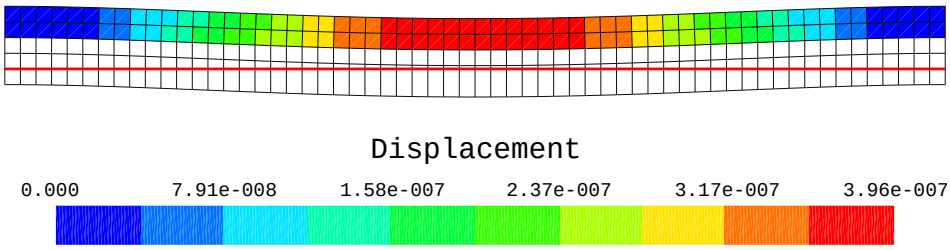



Figure 7.15: X-FEM pull-in position

Currently, resorting to X-FEM for this example is not very useful as the FEM mesh exhibits a good quality even at pull-in voltage. However, one can conceive that as long as the voltage increases the mesh gets more flat and singularities should occur at contact. Conversely, with X-FEM, the mesh remains good and continuing the analysis until the beam get in contact with the electrode could be considered. There is a great demand for such a capability from the companies involved in the design of MEMS devices.

Chapter 8

# Conclusions

## 8.1 Summary of the work

In this work, the eXtended Finite Element Method (X-FEM) is coupled with the Level Set description to investigate a novel approach to structural shape optimization. The X-FEM is used to avoid the mesh difficulties encountered with a CAD-FEM shape optimization by working with a fixed grid approach. Naturally associated to X-FEM, the Level Set description is used to describe the structural geometry and to provide the ability to modify the CAD model topology during the optimization process.

To this end, we have explored finite elements presenting internal void-material interface using X-FEM. The specific aspects of such elements have been presented in this manuscript with a focus on static and modal analyses of mechanical structures. The developments have been validated and a discussion on the accuracy of the method has been presented as well as comparison with FEM models and analytical solutions. One important issue that has been investigated is the quality of the X-FEM approximation for stresses. The source of the problems is the small elements that have only a very low fraction of their surface filled with material. These elements lead to large errors in the stress estimation and several approaches to circumvent or reduce this numerical error have been proposed. The problematic of embedded boundary conditions has also been succinctly treated. Imposition of Neumann boundary conditions has been presented and validated with analytical solutions. The Dirichlet boundary conditions have been successfully implemented using stabilized approaches. Moreover, a dedicated triangular element allowing strong Dirichlet conditions has been proposed.

To describe the geometry of structures, a Level Set description and a basic Level Set geometrical modeler have been developed. This tool allows the creation of basic shapes from analytical functions, simple geometrical features (circles, squares, ...) or more general curves and surfaces with a NURBS-to-Level Set converter. The concept of Constructive Solid Geometry with Level Set has been combined with the Level Set description and implemented to model complex 3D geometries. The Level Set description appears to be very attractive for shape optimization as this geometrical representation is able to treat merging interfaces as well as changes in the structure topology with a great flexibility.

Finally, the optimization module of OOFELIE has been extended to deal with the specifications

of these two new methods. The major contribution of this work lies in the investigation of the sensitivity analysis with the X-FEM method. Our developments have shown that resorting to a semi-analytic scheme may lead to large errors in certain configurations. Three approaches to circumvent these problems have been developed. The simplest strategy consists in adding a soft material in the void domain to keep a constant number of degrees of freedom. The second approach aims at adapting the perturbation step with a semi-analytic scheme. Finally, the last approach is to resort to an analytical sensitivity method that we have developed. We have shown that the computation of analytical derivatives can be realized with an approach similar to FEM. However, the expression of the structural matrices integrals and the nature of the shape parameters have a direct influence on the derivatives. It has been shown that this approach provides similar results with respect to semi-analytical or finite difference schemes. Several academic examples have illustrated the approach and showed that the sensitivity can be computed using domain or contour integrals (i.e. along the Level Set boundary). Finally, this approach has been extended to the case of element presenting enrichment functions such as the bi-material element.

The sensitivity analysis method has been developed for various objective functions and constraints such as displacements, compliance, volume, natural frequencies and local stresses. The novel shape optimization approach has been illustrated with several applications from classical shape optimization to industrial test cases. These applications have demonstrated the ability of X-FEM and Level Set to treat successfully a large range of applications including axisymmetric geometries, design dependent loads and self weight problems. Moreover, we have shown that topology modifications are possible using the proposed method. However, numerical applications have also illustrated that the geometrical parameterization of the model is very important in order to avoid invalid geometries.

The generalized shape optimization using X-FEM and Level Set description developed here takes place between shape and topology optimization. The X-FEM method takes full advantage of the fixed mesh work approach of topology optimization whereas the smooth curve description from the shape optimization is preserved. Contrary to shape optimization, no remeshing process is needed in our applications, and only one mesh is created for each test case. However, applications have shown that proceeding to shape optimization on a fixed grid presents some limitations and mesh refinement or adaptation may be necessary when the geometry undergoes large modifications. The various applications give encouraging results to study large scale industrial problems, especially complex 3D problems. However, by now, the present approach may be considered as an improved shape optimization method rather than a substitute to topology optimization. Therefore, future work should be devoted to extend the possibility to model deeper shape modifications and complex 3D geometries.

## 8.2 Achievements

This dissertation has tailored an novel shape optimization based on the Extended Finite Element Method coupled to a Level Set description of the geometry. The main contributions can be listed as follows:

- Implementation of an efficient C++ X-FEM code to perform mechanical and electrostatic analyses. The currently available developments contain 2D, 3D and axisymmetric elements with approximation orders up to the third degree;
- Development of a Constructive Solid Geometry modeler based on a Level Set description.

Regarding this topic, all lot of attention has been paid to design a generic tool with reliable and efficient mesh slicing routines;

- Investigation of different sensitivity analysis methods with X-FEM. A practical approach consisting in ignoring the change of the structural matrices dimension when using a semi-analytical scheme is proposed. Two alternative approaches are developed to reduce the error introduced with the *ignore* strategy. The first one consists in adapting the perturbation step to avoid the creation of new DOFs. The second one introduces a soft material in the void domain to prevent the creation of new DOFS. Moreover, a rigorous analytical sensitivity analysis is developed and validated on numerical applications for material-void and bi-material X-FEM elements. The analogy between FEM and X-FEM analytical sensitivity analyses is also highlighted;

- The problem of the evaluation of the stresses in small solid elements is exposed and several approaches to circumvent the problem have been investigated. To enhance the estimation of the stresses during the optimization, a geometrical mesh refinement criterion based on the Level Set is proposed;

- The developments have been validated on academic benchmarks and several applications considering 2D and 3D models as well as industrial application considering compliance, displacements, eigenfrequencies or stresses as objective functions and constraints have been realized. The capacity of the proposed structural optimization method to deal with topological modification is also demonstrated;

- To deal with design dependent loads, the prescription of Neumann boundary conditions has been considered. A novel enriched element is proposed to prescribe strongly the essential boundary conditions on triangular elements;

- Some pioneering work has been realized on the optimization of micro-structures, stochastic analysis under geometric uncertainties and electro-mechanical pull-in computation.

## 8.3 Future works

In this work, we have demonstrated the great potential of an optimization method coupling X-FEM with a Level Set description of the geometry. With the current implementation, various simple test cases have been realized. But it is a work in progress and we foresee several new features, which are necessary to get a complete framework able to treat a wider class of applications and industrial problems.

### 8.3.1 Level Set and CAD Modeling

Currently, the main limitation of our approach to tackle industrial applications lies in the creation of complex geometries. To our opinion, the development of a Level Set modeler makes no sense since it would take a lot of time and effort to develop a tool that would have equivalent robustness and features as existing commercial CAD tools. Another approach would be to interact with an existing CAD modeler in order to generate a Level Set model from the geometrical entities. However, most CAD commercial packages do not have a open interface (API) to provide access to the data structure of the geometrical model as well as the different relations and operations realized during the construction of the geometry. Therefore, even if this seems to be the best approach, in practice it is quite difficult to settle in practice. Furthermore, even if an

open interface is available, the implementation would be linked to a specific CAD package, it would thus be more interesting to work with standard CAD files.

In the chapter 2 we have shown that it is possible build a Level Set from a CAD file using a STL triangulation of the surfaces. However, this approach is not able to deal with complex geometries and especially with sharp corners. Hence, this approach should be extended in order to generate automatically high fidelity Level Set geometries from CAD files. This topic is currently studied by Mouton [123]. Current progress of this research indicates that complex geometries can be constructed automatically from B-Rep models. The introduction of these developments should be considered as well as the development of modification tools to be able to realize optimization problems.

### 8.3.2 Bi-material optimization

X-FEM elements to model bi-material interfaces have been introduced in OOFELIE in 2D and 3D and the expression of the analytical sensitivity for enriched elements has been developed in section 5.11. Currently, all the tools are readily available to study optimization of multiple materials models. One possible extension of the current development could be devoted to the optimization of stiffeners or inclusions.

### 8.3.3 Analytical velocity field of CAD entities and complex velocity fields

In this work, analytical sensitives have been implemented only for basic geometrical features such as circles, ellipsis, planes as those shapes present a simple analytical velocity field. When dealing with NURBS interfaces, the Velocity field is computed with a finite difference over the Gauss points movement. In this case, the procedure is not purely analytical because it involves a perturbation step. In [93], the authors show that the Velocity field can be obtained analytically if one has access to the parameterization of the curve. The same approach is detailed in [102, 158] for NURBS curves and surfaces. The introduction of a complete analytical method for this type of boundaries could therefore be considered.

### 8.3.4 Analysis of fixed CAD and image based models

In the industry, it is usual that companies have to work with models based on dead geometries. Generally, models are provided in a neutral geometrical format such as STEP and IGES or in a FEM mesh format. It is then very difficult to proceed to a shape optimization of the structure since the geometry is fixed. To overcome this problem, several authors have proposed [33] to modify directly the structural mesh using surface node coordinates and a suitable coupling between the nodal displacements or using mesh morphing approaches. These techniques are already implemented in commercial packages and perform quite well in practice. However, the number of variables can be very large and the modifications provided by these techniques can not alter the topology of the structure.

Our approach presents a major advantage over these techniques as one can modify the topology of the structure by adding a Level Set feature and optimize the shape of this component. Currently, our developments are limited to the insertion of holes or inclusion inside the mesh and a material domain can not be added. However, it can be conceived that using an enlarged meshed domain, the geometry could be modified and optimized. Hence, the extension of our approach

could provide a flexible method to study the influence of a geometrical change such as the insertion of an internal or external feature as well the modification of the boundaries or an insertion of defects. Coupling this approach with an efficient octree X-FEM approach would supply an efficient and fast tool to perform preliminary analysis of new designs. Finally, we should mention that several software packages are available to obtain a Level Set from a set of images. Analysis and optimization could therefore even be realized on structures without any CAD or initial mesh.

In the same topic, implementation of robust and accurate imposition of Dirichlet boundary conditions should be considered to be able to work with complete unconforming meshes.

### 8.3.5 Integrated layout design with X-FEM and Level Set

In his PhD thesis [184], Zhu proposed to couple topology and shape optimizations in order to perform an integrated layout design process. In his work, the main aim was to design new innovative geometries using a topology optimization approach that could take advantage of existing predefined moving structural objects inside the analysis domain.

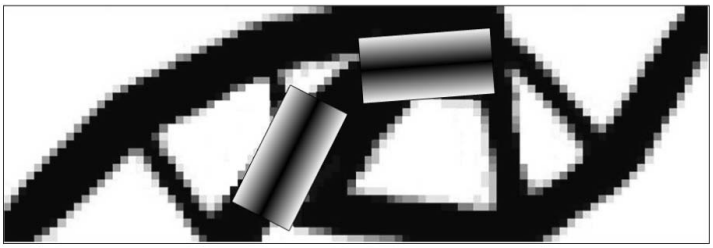


Figure 8.1: Simplified pylon optimal topology with two embedded objects [184]

Zhu proposed a successful two-step procedure composed of a topology optimization approach, to generate the global structure topology, followed by a shape optimization to optimize the positions of (possibly) multiple embedded objects as illustrated in Figure 8.1.

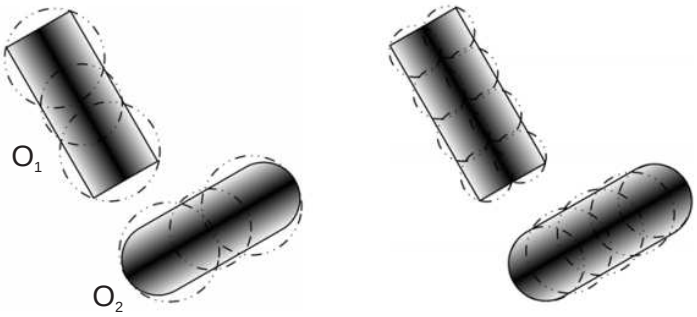



Figure 8.2: The FCM approximation of the components [184]

The main difficulty of the approach developed by Zhu relates to the prescription of non overlapping constraints between the different embedded objects. To this end, Zhu proposed to resort to the Finite Circle Method (FCM) that consists in approximating an object by overlapping its geometry with a finite number of circles as shown in Figure 8.2. The more circles we use to approximate an object, the higher is the accuracy of the geometry representation. The non

overlapping constraints of two objects $O_1$ and $O_2$ are then easily defined by computing distance between circles overlapping $O_1$ and $O_2$. The advantage of this approach being that both the distance constraints and their derivatives are continuous and easy to evaluate. However, when the embedded object becomes complex, one can imagine that the number of circles needed to obtain an acceptable geometrical overlapping can yield to a high number of constraints to evaluate.

In this application, we think that resorting to a Level Set Description for the several embedded objects could provide an interesting improvement to the FCM approach. Indeed, using one Level Set per embedded object, one design constraints for each pair of components is needed and the sensitivity of the constraint can be evaluated. The detection of overlapping objects is straightforward and call for a simple Level Set sign comparison.

# Bibliography


[1] Yazid Abdelaziz and Abdelmadjid Hamouine. A survey of the extended finite element. *Computers & Structures*, 86(11-12):1141 – 1151, 2008.

[2] David Adalsteinsson and James A. Sethian. A fast level set method for propagating interfaces. *J. Comput. Phys.*, 118(2):269–277, 1995.

[3] David Adalsteinsson and James A. Sethian. The fast construction of extension velocities in level set methods. *J. Comput. Phys.*, 148(1):2–22, 1999.

[4] Laurent Adam. Implémentation de classes de base nécessaires à l'optimisation de modèles éléments finis dans le logiciel oofelie. Technical report, University of Liège, 2003.

[5] Tekkaya A.E. and Güneri A. Shape optimization with the biological growth method: a parameter study. *Engng. Comp*, 13:4–18, 1998.

[6] Mehmet A. Akgün, John H. Garcelon, and Raphael T. Haftka. Fast exact linear and non-linear structural reanalysis and the sherman-morrison-woodbury formulas. *International Journal for Numerical Methods in Engineering*, 50(7):1587–1606, 2001.

[7] J. E. Akin. *Finite Element Analysis with Error Estimators.* Elsevier, draft edition, 2005.

[8] Grégoire Allaire, François Jouve, and Anca-Maria Toader. Structural optimization using sensitivity analysis and a level-set method. *Journal of Computational Physics*, 194(1):363–393, 2004.

[9] Grégoire Allaire, F. Jouve, and A. Toader. A level-set method for shape optimization. *C.R. Acad. Sci. Paris*, 334:1125–1130, 2002.

[10] Grégoire Allaire and François Jouve. Coupling the level set method and the topological gradient in structural optimization. In Martin Philip Bendsøe, Niels Olhoff, and Ole Sigmund, editors, *IUTAM Symposium on Topological Design Optimization of Structures, Machines and Materials*, volume 137 of *Solid Mechanics and Its Applications*, pages 3–12. Springer Netherlands, 2006.

[11] Altair. Optistruct. internet ressource : www.altair.com.

[12] Pedro M. A. Areias and T. Belytschko. A comment on the article 'a finite element method for simulation of strong and weak discontinuities in solid mechanics' by a. hansbo and p. hansbo [comput. methods appl. mech. engrg. 193 (2004) 3523-3540]. *Computer Methods in Applied Mechanics and Engineering*, 195(9-12):1275 – 1276, 2006.

[13] Helio J.C. Barbosa and Thomas J.R. Hughes. The finite element method with lagrange multipliers on the boundary: circumventing the babuska-brezzi condition. *Computer Methods in Applied Mechanics and Engineering*, 85(1):109 – 128, 1991.

[14] R. S. Barsoum. Triangular quarter-point elements as elastic and perfectly-plastic crack tip elements. *International Journal for Numerical Methods in Engineering*, 11(1):85 – 98, 1977.

[15] Éric Béchet, N. Moës, and Barbara Wohlmuth. A stable lagrange multiplier space for stiff interface conditions within the extended finite element method. *International Journal for Numerical Methods in Engineering*, 78(8):931–954, 2009.

[16] E. Béchet, H. Minnebo, N. Moës, and B. Burgardt. Improved implementation and robustness study of the x-fem for stress analysis around cracks. *International Journal for Numerical Methods in Engineering*, 64:1033–1056, 2005.

[17] Pierre Beckers. Recent developments in shape sensitivity. *Engineering Optimization*, 18:67–78, 1991.

[18] A. D. Belgundu and S. D. Rajan. Shape optimization approach based on natural design variables and shape functions. *Computer Methods in applied mechanics and engineering*, 66:87–106, 1988.

[19] T. Belytschko and T. Black. Elastic crack growth in finite elements with minimal remeshing. *International Journal for Numerical Methods in Engineering*, 45:601–620, 1999.

[20] T. Belytschko and M. Fleming. Smoothing, enrichment and contact in the element-free galerkin method. *Computers & Structures*, 71(2):173 – 195, 1999.

[21] T. Belytschko, Robert Gracie, and Giulio Ventura. A review of extended/generalized finite element methods for material modeling. *Modelling and Simulation in Materials Science and Engineering*, 17(4):043001+, 2009.

[22] T. Belytschko, Y. Y. Lu, and L. Gu. Element-free galerkin methods. *International Journal for Numerical Methods in Engineering*, 27(2):229 – 256, 1994.

[23] T. Belytschko, Y. Y. Lu, and L. Gu. Crack propagation by element-free galerkin methods. *Engineering Fracture Mechanics*, 51(2):295 – 315, 1995.

[24] T. Belytschko, N. Moës, S. Usui, and C. Parimi. Arbitrary discontinuities in finite elements. *International Journal for Numerical Methods in Engineering*, 50:993–1013, 2001.

[25] T. Belytschko, Chandu Parimi, N. Moës, N. Sukumar, and Shuji Usui. Structured extended finite element methods for solids defined by implicit surfaces. *International Journal for Numerical Methods in Engineering*, 56:609–635, 2003.

[26] T. Belytschko, S. P. Xiao, and C. Parimi. Topology optimization with implicit functions and regularization. *International Journal for Numerical Methods in Engineering*, 57:1177–1196, 2003.

[27] M. P. Bendsøe. Optimal shape design as a material distribution problem. *Journal of Structural Optimization*, 1:193–202, 1989.

[28] Martin Philip Bendsoe and Noboru Kikuchi. Generating optimal topologies in structural design using a homogenization method. *Computer Methods in Applied Mechanics and Engineering*, 71(2):197–224, 1988.

[29] Martin Philip Bendsoe and Ole Sigmund. *Topology Optimization: Theory, Methods, and Applications*. Springer Verlag, 2003.

[30] J. A. Bennet and M. E. Botkin. Structural shape optimization with geometric description and adaptive mesh refinement. *AIAA*, 23:458–464, 1984.

[31] E. Benvenuti, A. Tralli, and G. Ventura. A regularized xfem model for the transition from continuous to discontinuous displacements. *International Journal for Numerical Methods in Engineering*, 74(6):911–944, 2008.

[32] Christian Bischof, Peyvand Khademi, Andrew Mauer, and Alan Carle. Adifor 2.0: Automatic differentiation of fortran 77 programs. *Computing in Science and Engineering*, 3:18–32, 1996.

[33] Kai-Uwe Bletzinger, Matthias Firl, Johannes Linhard, and Roland Wüchner. Optimal shapes of mechanically motivated surfaces. *Computer Methods in Applied Mechanics and Engineering*, 199(5Ũ8):324 – 333, 2010.

[34] S. Bordas, P. V. Nguyen, C. Dunant, A. Guidoum, and H. Nguyen-Dang. An extended finite element library. *International Journal for Numerical Methods in Engineering*, 71:703 – 732, 2007.

[35] Stéphane Bordas, Marc Duflot, and Phong Le. A simple error estimator for extended finite elements. *Communications in Numerical Methods in Engineering*, 24(11):961–971, 2008.

[36] V. Braibant and P. Morelle. Shape optimal design and free mesh generation. *Structural Optimization*, 2:223–231, 1990.

[37] Vincent Braibant. *Optimisation de forme des structures en vue de la conception assistée par ordinateur*. PhD thesis, University of Liège, 1986.

[38] Vincent Braibant and Claude Fleury. Shape optimal design using b-splines. *Computer Methods in Applied Mechanics and Engineering*, 44:247–267, 1984.

[39] G. Bricteux, P Dular, P Duysinx, C Geuzaine, J.-F. Remacle, R Sabariego, and L. Van Miegroet. Technical report efconivo, January 2012.

[40] M. Bruyneel and P. Duysinx. Note on topology optimization of continuum structures including self-weight. *Structural and Multidisciplinary Optimization*, 29:245–256, 2005.

[41] M.D. Buhmann. *Radial Basis Functions: Theory and Implementations*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, 2009.

[42] Jiaqin Chen, Vadim Shapiro, Krishnan Suresh, and Igor Tsukanov. Shape optimization with topological changes and parametric control. *International Journal for Numerical Methods in Engineering*, 71(3):313–346, 2007.

[43] Kwok Wah Cheng and T.-P. Fries. Higher-order xfem for curved strong and weak discontinuities. *International Journal for Numerical Methods in Engineering*, 82(5):564–590, 2010.

[44] J. Chessa and T. Belytschko. An extended finite element method for two-phase fluids. *Journal of Applied Mechanics*, 70:10–17, 2003.

[45] J. Chessa, P. Smolinski, and T. Belytschko. The extended finite element method for solidification problems. *International Journal for Numerical Methods in Engineering*, 53:1959–1977, 2002.

[46] Jack Chessa, Hongwu Wang, and T. Belytschko. On the construction of blending elements for local parition of unity enriched finite elements. Technical report, Northwestern University, November 2002.

[47] K.K. Choi and N.H. Kim. *Structural Sensitivity Analysis and Optimization: Linear systems*. Mechanical engineering series. Springer Science+Business Media, 2005.

[48] David L. Chopp. Computing minimal surfaces via level set curvature flow. *J. Comput. Phys.*, 106(1):77–91, 1993.

[49] R. D. Cook, D. S. Malkus, M. E. Plesha, and R. J. Witt. *Concepts and applications of finite element analysis*. John Wiley and Sons. Inc., fourth edition, 2002.

[50] J. Dankova and J. Haslinger. Numerical realization of a fictitious domain approach used in shape optimization. part i distributed controls. *Applications of Mathematics*, 41:123–147, 1996.

[51] C. Daux, N. Moës, J. Dolbow, N. Sukumar, and T. Belytschko. Arbitrary branched and intersecting cracks with the extended finite element method. *International Journal for Numerical Methods in Engineering*, 48:1741–1760, 2000.

[52] John Dolbow, N. Moës, and T. Belytschko. Discontinuous enrichment in finite elements with a partition of unity method. *Finite Elements in Analysis and Design*, 36(3-4):235 – 260, 2000.

[53] K. Dréau, N. Chevaugeon, and N. Moës. Studied x-fem enrichment to handle material interfaces with higher order finite element. *Computer Methods in Applied Mechanics and Engineering*, 199(29-32):1922 – 1936, 2010.

[54] Marc Duflot and Stéphane Bordas. A posteriori error estimation for extended finite elements by an extended global recovery. *International Journal for Numerical Methods in Engineering*, 76(8):1123–1138, 2008.

[55] P. Duysinx, E. Lemaire, and L Van Miegroet. Application de l'optimisation de forme et de topologie a la conception innovante d'ejecteurs pour un concasseur mag'impact, December 2009.

[56] P. Duysinx, W.H. Zhang, H.G. Zhong, P. Beckers, and C. Fleury. Structural shape optimization with error control", 20th asme design automation conference. In *Mineapolis (Minesota-USA), 11-14 septembre, 1994.*, 1994.

[57] Pierre Duysinx. *Optimisation Topologique : du Milieu Continu à la Structure Elastique.* PhD thesis, University of Liège, 1996.

[58] Pierre Duysinx and Martin Philip Bendsoe. Control of local stresses in topology optimization of continuum strucutres. *International Journal for Numerical Methods in Engineering*, 43:1453–1478, 1998.

[59] Douglas Enright, Stephen Marschner, and Ronald Fedkiw. Animation and rendering of complex water surfaces. *ACM Trans. Graph.*, 21:736–744, July 2002.

[60] Gerald Farin. *Curves and surfaces for CAGD: a practical guide.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 5th edition, 2002.

[61] Carlos A. Felippa. *Introduction to Finite Element Methods.* PhD thesis, Department of Aerospace Egineering Sciences, University of Colorado at Boulder, 2005.

[62] Claude Fleury. Méthodes numériques d'optimisation des structures. *rapport interne LTAS*, SF-19, 1973.

[63] Claude Fleury. Structural weight optimization by dual methods of convex programming. *International Journal for Numerical Methods in Engineering*, 14(12):1761–1783, 1979.

[64] Claude Fleury. An efficient optimality criteria approach to the minimum weight design of elastic structures. *Computers & Structures*, 11(3):163 – 173, 1980.

[65] Claude Fleury. Conlin: An efficient dual optimizer based on convex approximation concepts. *Structural and Multidisciplinary Optimization*, 1:81–89, Jun 1989.

[66] Claude Fleury. Dual methods for convex seperable problems", dans: Optimization of large structural systems. In G.I.N. Rozvany, editor, *8th International Symposium on Automated Technology for Verification and Analysis (ATVA 2010)*, volume 231 of *NATO ASI Series*, pages 509–530. Kluwer Academic Publishers, 1993.

[67] Claude Fleury. Sequential convex programming for structural optimization problems. *Optimization of large structural systems*, 1:531–553, 1993.

[68] R.L. Fox. Constraint surface normals for structural synthesis techniques. *AIAA Journal, vol.*, 8:1517 – 1518, 1965.

[69] T.-P. Fries and T. Belytschko. The intrinsic xfem: a method for arbitrary discontinuities without additional unknowns. *International Journal for Numerical Methods in Engineering*, 9999:n/a, 2006.

[70] T.-P. Fries and T. Belytschko. The extended/generalized finite element method: An overview of the method and its applications. *International Journal for Numerical Methods in Engineering*, 84(3):253 – 304, 2010.

[71] T.-P. Fries, N. Moës, and T. Belytschko. A corrected xfem approximation without problems in blending elements. *International Journal for Numerical Methods in Engineering*, 75(5):503 – 532, 2008.

[72] M.B. Fuchs and N.N.Y. Shemesh. Density-based topological design of structures subjected to water pressure using a parametric loading surface. *Structural and Multidisciplinary Optimization*, 28:11–19, 2004.

[73] Stéphane Garreau, Philippe Guillaume, and Mohamed Masmoudi. The topological asymptotic for pde systems: The elasticity case. *SIAM J. Control Optim.*, 39(6):1756–1778, 2000.

[74] Axel Gerstenberger and Wolfgang A. Wall. An extended finite element method/lagrange multiplier based approach for fluid-structure interaction. *Computer Methods in Applied Mechanics and Engineering*, 197(19-20):1699 – 1714, 2008.

[75] Christophe Geuzaine and Jean-François Remacle. Gmsh: A 3-d finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, 79(11):1309–1331, 2009.

[76] Andreas Griewank, David Juedes, and Jean Utke. Algorithm 755: ADOL-C: a package for the automatic differentiation of algorithms written in C/C++. *ACM Transactions on Mathematical Software*, 22:131–167, 1996.

[77] Seung-Hyun Ha, K. Choi, and Seonho Cho. Numerical method for shape optimization using t-spline based isogeometric method. *Structural and Multidisciplinary Optimization*, 42:417–428, 2010.

[78] Seung-Hyun Ha, K. Choi, and Seonho Cho. Numerical method for shape optimization using t-spline based isogeometric method. *Structural and Multidisciplinary Optimization*, 42:417–428, 2010.

[79] R T Haftka and R V Grandhi. Structural shape optimization - a survey. *Comput. Methods Appl. Mech. Eng.*, 57:91–106, August 1986.

[80] R. T. Haftka and Z. Gürdal. *Elements of Structural Optimization*. Kluwer, 3rd edition, 1993.

[81] R.T. Haftka and H.M. Adelman. Sensitivity of discrete systems. In *Optimization of Large Structural Systems, (G.I.N. Rozvany, éd.), Vol. I, pp 289-311, 1993, NATO ASI Series, Vol. 231, 1993, Kluwer Academic Publishers.*, 1993.

[82] Jaroslav Haslinger and Yves Renard. A new fictitious domain approach inspired by the extended finite element method. *SIAM Journal on Numerical Analysis*, 47(2):1474–1499, 2009.

[83] T.J.R. Hughes, J.A. Cottrell, and Y. Bazilevs. Isogeometric analysis: Cad, finite elements, nurbs, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering*, 194(39-41):4135 – 4195, 2005.

[84] Babuska I. and Melenk J. M. The partition of unity method. *International Journal for Numerical Methods in Engineering*, 40:727–758, 1997.

[85] E. V. Iarve. Mesh independent modelling of cracks by using higher order shape functions. *Int. J. Numer. Meth. Engng.*, 56(6):869–82, 2003.

[86] Sergio R. Idelsohn, Eugenio Oñate, Nestor Calvo, and Facundo Del Pin. The meshless finite element method. *International Journal for Numerical Methods in Engineering*, 58:893–912, 2003.

[87] Gang Won Jang, Nico van Dijk, and Fred van Keulen. Level-set-based topology optimization of mems considering etching uncertainties. In *$8^{th}$ World Congress of Structural and Multidisciplinary Optimization*, 2009.

[88] H. Ji and J. E. Dolbow. On strategies for enforcing interfacial constraints and evaluating jump conditions with the extended finite element method. *International Journal for Numerical Methods in Engineering*, 61:2508–2535, 2004.

[89] Guang-Shan Jiang and Danping Peng. Weighted eno schemes for hamilton–jacobi equations. *SIAM J. Sci. Comput.*, 21(6):2126–2143, 1999.

[90] B. L. Karihaloo and Q. Z. Xiao. Modelling of stationary and growing cracks in fe framework without remeshing: a state-of-the-art review. *Computers & Structures*, 81(3):119 – 129, 2003.

[91] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International Journal Of Compuer Vision*, 1(4):321–331, 1988.

[92] Andreas Keese. A review of recent developments in the numerical solution of stochastic partial differential equations ( stochastic finite elements ) a review of recent developments in the numerical solution of stochastic partial differential equations ( stochastic finit. *October*, 2003.

[93] Nam H. Kim and Youngmin Chang. Eulerian shape design sensitivity analysis and optimization with a fixed grid. *Computer Methods in Applied Mechanics and Engineering*, 194(30-33):3291 – 3314, 2005.

[94] N.H. Kim, K.K. Choi, and M.E. Botkin. Numerical method for shape optimization using meshfree method. *Structural and Multidisciplinary Optimization*, 24:418–429, 2002.

[95] P. Krysl and T. Belytschko. The element free galerkin method for dynamic propagation of arbitrary 3-d cracks. *International Journal for Numerical Methods in Engineering*, 44(6):767 – 800, 1999.

[96] P. Laborde, J. Pommier, and M. Salaün Y. Renard. High-order extended finite element method for cracked domains. *International Journal for Numerical Methods in Engineering*, 64:354–381, 2005.

[97] A. Legay, J. Chessa, and T. Belytschko. An eulerian-lagrangian method for fluid-structure interaction based on level sets. *Computer Methods in Applied Mechanics and Engineering*, 195(17-18):2070 – 2087, 2006.

[98] A. Legay, H. W. Wang, and T. Belytschko. Strong and weak arbitrary discontinuities in spectral finite elements. *International Journal for Numerical Methods in Engineering*, 64:991–1008, 2005.

[99] Séverine Lepage. *Stochastic Finite Element Method for the Modeling of Thermoelastic Damping in Micro-Resonators*. PhD thesis, University of Liège, 2007.

[100] Z. Li. The immersed interface method - a numerical approach for partial differential equations with interfaces, 1994.

[101] D. Liefooghe and C. Fleury. An interactive capability for shape optimal design. *Finite Elem. Anal. Des.*, 5:39–55, April 1989.

[102] T. Lindby and J. L. T. Santos. 2-d and 3-d shape optimization using mesh velocities to integrate analytical sensitivities with associative cad. *Structural and Multidisciplinary Optimization*, 13:213–222, 1997.

[103] W. K. Liu, S. Jun, and Y. F. Zhang. Reproducing kernel particle methods. *International Journal for Numerical Methods in Fluids*, 20(8-9):1081 – 1106, 1995.

[104] Xu-Dong Liu, Stanley Osher, and Tony Chan. Weighted essentially non-oscillatory schemes. *Journal of Computational Physics*, 115(1):200 – 212, 1994.

[105] R. Malladi and J. A. Sethian. A unified approach to noise removal, image enhancement, and shape recovery. *IEEE Trans. On Image Processing*, 5:1554–1568, 1996.

[106] Ravi Malladi and James A. Sethian. Image processing: Flows under min/max curvature and mean curvature. *CVGIP: Graphical Model and Image Processing*, 58(2):127–141, 1996.

[107] V. Mallet, D.E. Keyes, and F.E. Fendell. Modeling wildland fire propagation with level set methods. *Computers & Mathematics with Applications*, 57(7):1089 – 1101, 2009.

[108] Ursula M. Mayer, Axel Gerstenberger, and Wolfgang A. Wall. Interface handling for three-dimensional higher-order xfem-computations in fluid-structure interaction. *International Journal for Numerical Methods in Engineering*, 79:846–869, 2009.

[109] Shaker A. Meguid. *Integrated Computer-Aided Design of Mechanical Systems*. Springer;, 1986.

[110] J. M. Melenk and I. Babuska. The partition of unity finite element method: Basic theory and applications. *Computer Methods in Applied Mechanics and Engineering*, 139:289–314, 1996.

[111] Ian M. Mitchell. The flexible, extensible and efficient toolbox of level set methods. *J. Sci. Comput.*, 35(2-3):300–329, 2008.

[112] N. Moës, E. Béchet, and M. Tourbier. Imposing dirichlet boundary conditions in the extended finite element method. *International Journal for Numerical Methods in Engineering*, 67:n/a, 2006.

[113] N. Moës, M. Cloirec, P. Cartraud, and J. F. Remacle. A computational approach to handle complex microstructure geometries. *Computer Methods in Applied Mechanics and Engineering*, 192:3163–3177, Jul 2003.

[114] N. Moës, John Dolbow, and T. Belytschko. A finite element method for crack growth without remeshing. *International Journal for Numerical Methods in Engineering*, 46:131–150, 1999.

[115] N. Moës, A. Gravouil, and T. Belytschko. Non-planar 3d crack growth by the extended finite element and level sets - part i: Mechanical model. *International Journal for Numerical Methods in Engineering*, 53:2549–2568, 2002.

[116] S. Mohammadi. *Extended Finite Element Method for Fracture Analysis of Structures.* Blackwell, 2008.

[117] A.J. Morris. *Foundations of Structural Optimization: A Unified Approach.* Wiley & Sons, 1982.

[118] B Morse, T Yoo, P Rheingans, and D Chen. Interpolating implicit surfaces from scattered surface data using compactly supported radial. *Proceedings of the International Conference on Shape*, 2001.

[119] C. D. Mote. Global-local finite element. *International Journal for Numerical Methods in Engineering*, 3(4):565 – 574, 1971.

[120] David M. Mount and Sunil Arya. Ann library: Approximate nearest neighbor searching. internet resources: http://www.cs.umd.edu/ mount/ann/.

[121] Hashem M. Mourad and Isaac Harari John Dolbow. A bubble-stabilized finite element method for dirichlet constraints on embedded interfaces. *International Journal for Numerical Methods in Engineering*, 9999:n/a, 2006.

[122] S. E. Mousavi, H. Xiao, and N. Sukumar. Generalized gaussian quadrature rules on arbitrary polygons. *International Journal for Numerical Methods in Engineering*, 82:99–113, 2010.

[123] T. Mouton and E. Béchet. Conversion of a b-rep cad model to an implicit representation in the context of the x-fem. In *Advanced COmputational Methods in ENgineering, 14-17 November 2011, University of Liège, Liège, Belgium*, 2011.

[124] W. Mulder, S. Osher, and James A. Sethian. Computing interface motion in compressible gas dynamics. *J. Comput. Phys.*, 100:209–228, June 1992.

[125] S. Natarajan, S. Bordas, and D. Roy Mahapatra. Numerical integration over arbitrary polygonal domains based on schwarz-christoffel conformal mapping. *International Journal for Numerical Methods in Engineering*, 80(1):103–134, 2009.

[126] S. Natarajan, D. R. Mahapatra, and S. P. A. Bordas. Integrating strong and weak discontinuities without integration subcells and example applications in an xfem/gfem framework. *International Journal for Numerical Methods in Engineering*, 83(3):269–294, 2010.

[127] J. Norato, R. Haber, D. Tortorelli, and M. P. Bendsøe. A geometry projection method for shape optimization. *International Journal for Numerical Methods in Engineering*, 60:2289–2312, 2004.

[128] A. A. Novotny, R. A. Feijoo, E. Taroco, and C. Padra. Topological sensitivity analysis. *Computer Methods in Applied Mechanics and Engineering*, 192:803–829, Feb 2003.

[129] Yutaka Ohtake, Alexander Belyaev, and Hans-Peter Seidel. *Multi-scale and Adaptive CS-RBFs for Shape Reconstruction from Cloud of Points*, pages 143–154. Springer, Berlin, Germany, 2005.

[130] Ole and Sigmund. Materials with prescribed constitutive parameters: An inverse homogenization problem. *International Journal of Solids and Structures*, 31(17):2313 – 2329, 1994.

[131] Niels Olhoff, Martin P. Bendsøe, and John Rasmussen. On cad-integrated structural topology and design optimization. *Computer Methods in Applied Mechanics and Engineering*, 89(1-3):259–279, 1991.

[132] Oofelie. an object oriented finite element code led by interactive executor. internet resources: www.open-engineering.com.

[133] Stanley Osher and James A Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, 79:12–49, 1988.

[134] Stanley Osher and Chi-Wang Shu. High-order essentially nonsocillatory schemes for hamilton-jacobi equations. *SIAM J. Numer. Anal.*, 28(4):907–922, 1991.

[135] Stanley J. Osher and Ronald P. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Springer, 1 edition, October 2002.

[136] Niels L. Pedersen. On topology optimization of plates with prestress. *International Journal for Numerical Methods in Engineering*, 51(2):225–239, 2001.

[137] Pauli Pedersen. *Optimal Designs - Structures and Materials - Problems and Tools*. DTU Mechanical Engineering, 2003.

[138] Walter D. Pilkey. *Peterson's Stress Concentration Factors, 2nd Edition*. Wiley and Sons, 1997.

[139] W. Prager and J.E. Taylor. Problems of optimal structural design. *Journal of Applied Mechanics*, 35:102–106, 1968.

[140] T. Rabczuk, S. Bordas, and G. Zi. On three-dimensional modelling of crack growth using partition of unity methods. *Computers & Structures*, 88(23-24):1391 – 1411, 2010.

[141] T. Rabczuk, S. P. Xiao, and M. Sauer. Coupling of mesh-free methods with finite elements: basic concepts and test results. *Communications in Numerical Methods in Engineering*, 22(10):1031 – 1065, 2006.

[142] Y. Radovcic and A. Remouchamps. Boss quattro: an open system for parametric design. *Structural and Multidisciplinary Optimization*, 23:140Ũ152, 2002.

[143] Véronique Rochus. *Finite Element Modelling of Strong Electro-Mechanical Coupling in MEMS*. PhD thesis, University of Liège, February 2006.

[144] Véronique Rochus, Laurent Van Miegroet, Daniel J. Rixen, and Pierre Duysinx. Electrostatic simulation using xfem for conductor and dielectric interfaces. *International Journal for Numerical Methods in Engineering*, 85(10):1207–1226, 2011.

[145] Martin H. Sadd. *Elasticity theory, applications, and numerics*. Academic Press, 2009.

[146] Samtech. Topol. internet ressource : www.samtech.be.

[147] G. N. Savin. *Stress concentration around holes*. Pergamon Press, New York, 1961.

[148] L.A. Schmit. Structural design by systematic synthesis. *Proceedings of the 2nd ASCE Conference on Electronic Computation, New York*, pages 105 – 132, 1960.

[149] L.A. Schmit and B. Farshi. Some approximation concepts for structural synthesis. *AIAA Journal*, 12:692–699, 1974.

[150] L.A. Schmit and H. Miura. Approximation concepts for efficient structural synthesis. *NASA Contractor Report NASA-CR 2552*, 1976.

[151] Thibaut Schoonjans. Application of x-fem and level-set description to shape optimization of holes in elastic plates and periodic microstructures, June 2009.

[152] Stephen D Senturia. *Microsystem Design*. Kluwer Academic Publishers, 2001.

[153] J. A. Sethian and A. Wiegmann. Structural boundary design via level set and immersed interface methods. *Journal of Computational Physics*, 163:489–528, 2000.

[154] James Albert Sethian. *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision and Materials Science*. Cambridge Monograph on Applied and Computational Mathematics. Cambridge University Press, 1999.

[155] A. P. Seyranian, E. Lund, and N. Olhoff. Multiple eigenvalues in structural optimization problems. *Structural and Multidisciplinary Optimization*, 8:207–227, 1994.

[156] Vadim Shapiro and Igor Tsukanov. Implicit functions with guaranteed differential properties. In *Proceedings of the fifth ACM symposium on Solid modeling and applications*, SMA '99, pages 258–269, New York, NY, USA, 1999. ACM.

[157] Shmuel and Vigdergauz. The effective properties of a perforated elastic plate numerical optimization by genetic algorithm. *International Journal of Solids and Structures*, 38(48Ű49):8593 – 8616, 2001.

[158] C. Silva and M. Bittencourt. Velocity fields using nurbs with distortion control for structural shape optimization. *Structural and Multidisciplinary Optimization*, 33:147–159, 2007.

[159] Peter Smereka. Spiral crystal growth. *Physica D*, 138:282–301, 2000.

[160] J. Sokolowski and A. Zochowski. On the topological derivative in shape optimization. *SIAM J. Control Optim.*, 37(4):1251–1272, 1999.

[161] F. L. Stazi, E. Budyn, J. Chessa, and T. Belytschko. An extended finite element method with higher-order elements for curved cracks. *Computational Mechanics*, 31:38–48, 2003.

[162] M. Stolarska, D. L. Chopp, N. Moës, and T. Belytschko. Modelling crack growth by level sets in the extended finite element method. *International Journal for Numerical Methods in Engineering*, 51:943–960, 2001.

[163] T. Strouboulis, I. Babuska, and K. Copps. The design and analysis of the generalized finite element method. *Computer Methods in Applied Mechanics and Engineering*, 181(1-3):43 – 69, 2000.

[164] N. Sukumar, D. L. Chopp, N. Moes, and T. Belytschko. Modeling holes and inclusions by level sets in the extended finite-element method. *Computer Methods in Applied Mechanics and Engineering*, 190:6183–6200, 2001.

[165] N. Sukumar, D. L. Chopp, and B. Moran. Extended finite element method and fast marching method for three-dimensional fatigue crack propagation. *Engineering Fracture Mechanics*, 70:29–48, 2003.

[166] N. Sukumar, N. Moës, B. Moran, and T. Belytschko. Extended finite element method for three-dimensional crack modelling. *International Journal for Numerical Methods in Engineering*, 48:1549–1570, 2000.

[167] Mark Sussman, Peter Smereka, and Stanley Osher. A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational Physics*, 114(1):146–159, 1994.

[168] Ivan E. Sutherland, Robert F. Sproull, and Robert A. Schumacker. A characterization of ten hidden-surface algorithms. *ACM Comput. Surv.*, 6(1):1–55, 1974.

[169] Barna Szabó and Ivo Babuška. *Finite Element Analysis*. John Wiley and Sons, Ltd., New York, London, Sydney, 1991.

[170] S. Timoshenko and J. N. Goodier. *Theory of Elasticity*. McGraw-Hill, Inc. New York, 1951.

[171] Greg Turk and James F. O'Brien. Modelling with implicit surfaces that interpolate. *ACM Transactions on Graphics*, 21(4):855–873, October 2002.

[172] F. van Keulen, R.T. Haftka, and N.H. Kim. Review of options for structural design sensitivity analysis. part 1: Linear systems. *Computer Methods in Applied Mechanics and Engineering*, 194(30-33):3213 – 3243, 2005.

[173] L. Van Miegroet. Sensitivity analysis of structural shape using x-fem and level set description, September 2006.

[174] Laurent Van Miegroet. Optimisation de structures basée sur la méthode des courbes de niveau. Technical report, University of Liège, 2004.

[175] G. Ventura. On the elimination of quadrature subcells for discontinuous functions in the extended finite-element method. *Int. J. Numer. Meth. Engng*, 66:761–795, 2006.

[176] Michael Yu Wang and Xiaoming Wang. Color level sets: a multi-phase method for structural topology optimization with multiple materials. *Computer Methods in Applied Mechanics and Engineering*, 193:469–496, Feb 2004.

[177] Michael Yu Wang, Xiaoming Wang, and Dongming Guo. A level set method for structural topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 192:227–246, Jan 2003.

[178] Michael Yu Wang, Xiaoming Wang, and Dongming Guo. A level set method for structural topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 192(1-2):227 – 246, 2003.

[179] Shengyin Wang and Michael Yu Wang. Radial basis functions and level set method for structural topology optimization. *International Journal for Numerical Methods in Engineering*, 65(12):2060–2090, 2006.

[180] Qi Xia, Tielin Shi, and Michael Wang. A level set based shape and topology optimization method for maximizing the simple or repeated first eigenvalue of structure vibration. *Structural and Multidisciplinary Optimization*, 43:473–485, 2011.

[181] Qi Xia and Michael Wang. Topology optimization of thermoelastic structures using level set method. *Computational Mechanics*, 42:837–857, 2008.

[182] L. Yatziv, A. Bartesaghi, and G. Sapiro. O(N) implementation of the fast marching algorithm. *Journal of Computational Physics*, 212:393–399, March 2006.

[183] Wei-Hong Zhang. *Calcul des sensibilités et optimization de frome par la méthode des éléments finis*. PhD thesis, Université de Liège, 1992.

[184] Jihong Zhu. *Integrated Layout Design of Multi-component Systems*. PhD thesis, University of Liège, Northwestern Polytechnical University of Xian (NPU), 2008.

[185] Goangseup Zi and T. Belytschko. New crack-tip elements for xfem and applications to cohesive cracks. *International Journal for Numerical Methods in Engineering*, 57:2221–2240, 2003.

[186] O. C. Zienkiewicz and R. L. Taylor. *The Finite Element Method, Volume 1: Basic Formulation and Linear Problems*. McGraw-Hill, London, 1989.

[187] O. C. Zienkiewicz and R. L. Taylor. *Finite Element Method: Volume 2, Solid Mechanics (Finite Element Method Series)*. Butterworth-Heinemann, 5 edition, September 2000.

[188] O. C. Zienkiewicz and J. Z. Zhu. The superconvergent patch recovery anda posteriori error estimates. part 1: The recovery technique. *International Journal for Numerical Methods in Engineering*, 33:1331–1364, 1992.